

FIG. 1 Error Recovery Architecture

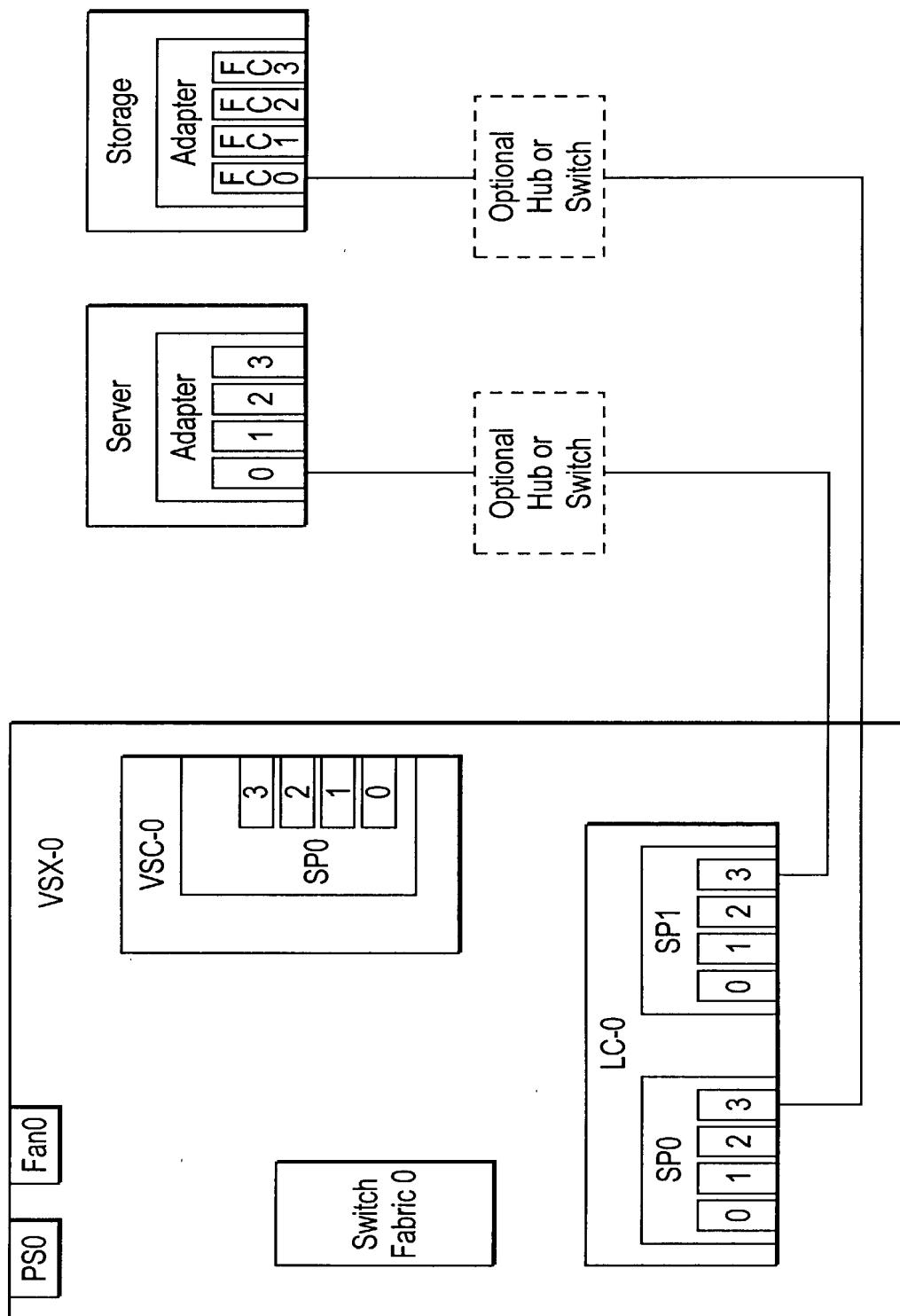


FIG. 2 Non-Fault Tolerant Configuration

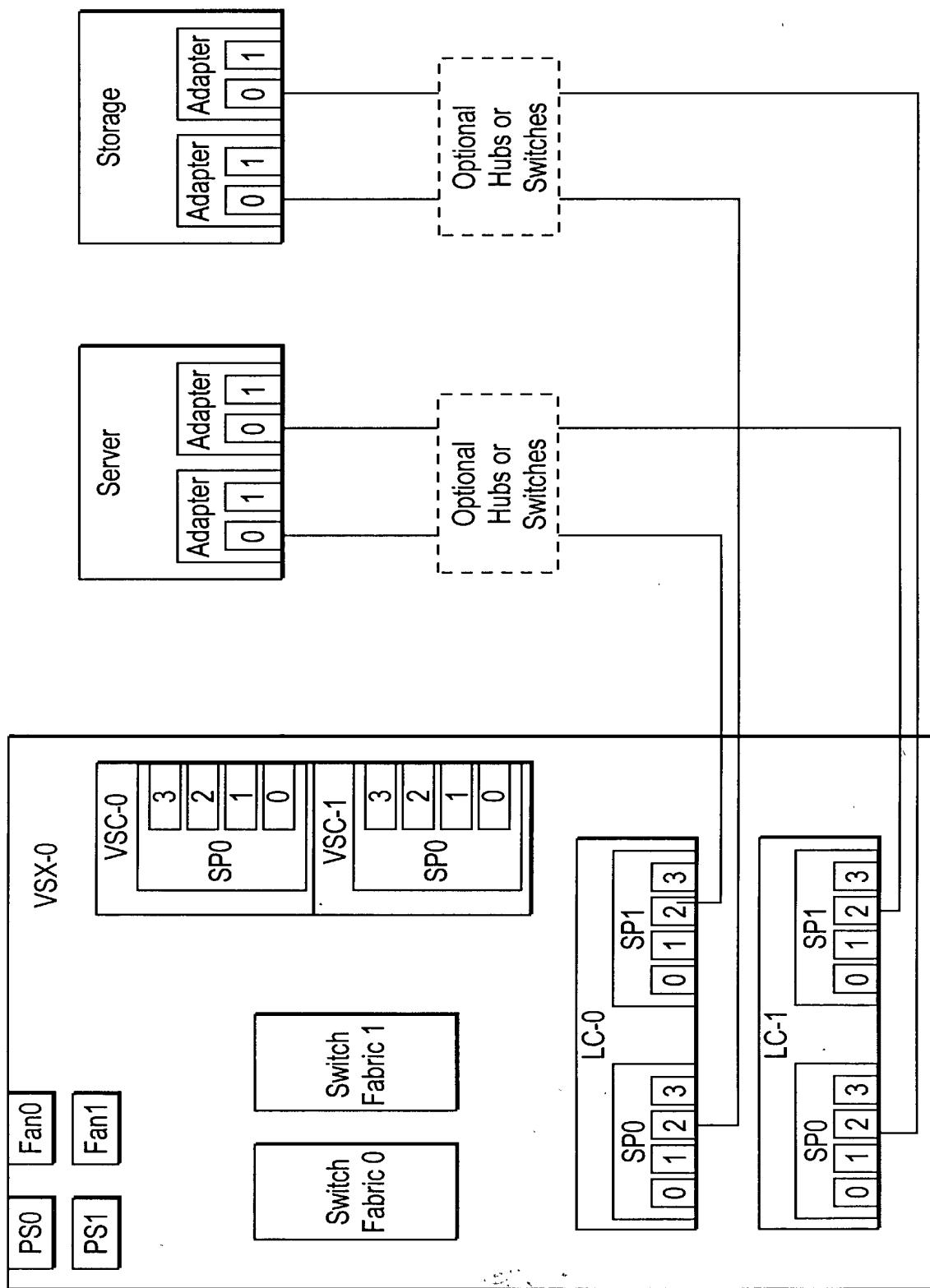


FIG. 3 Fault Tolerant Configuration

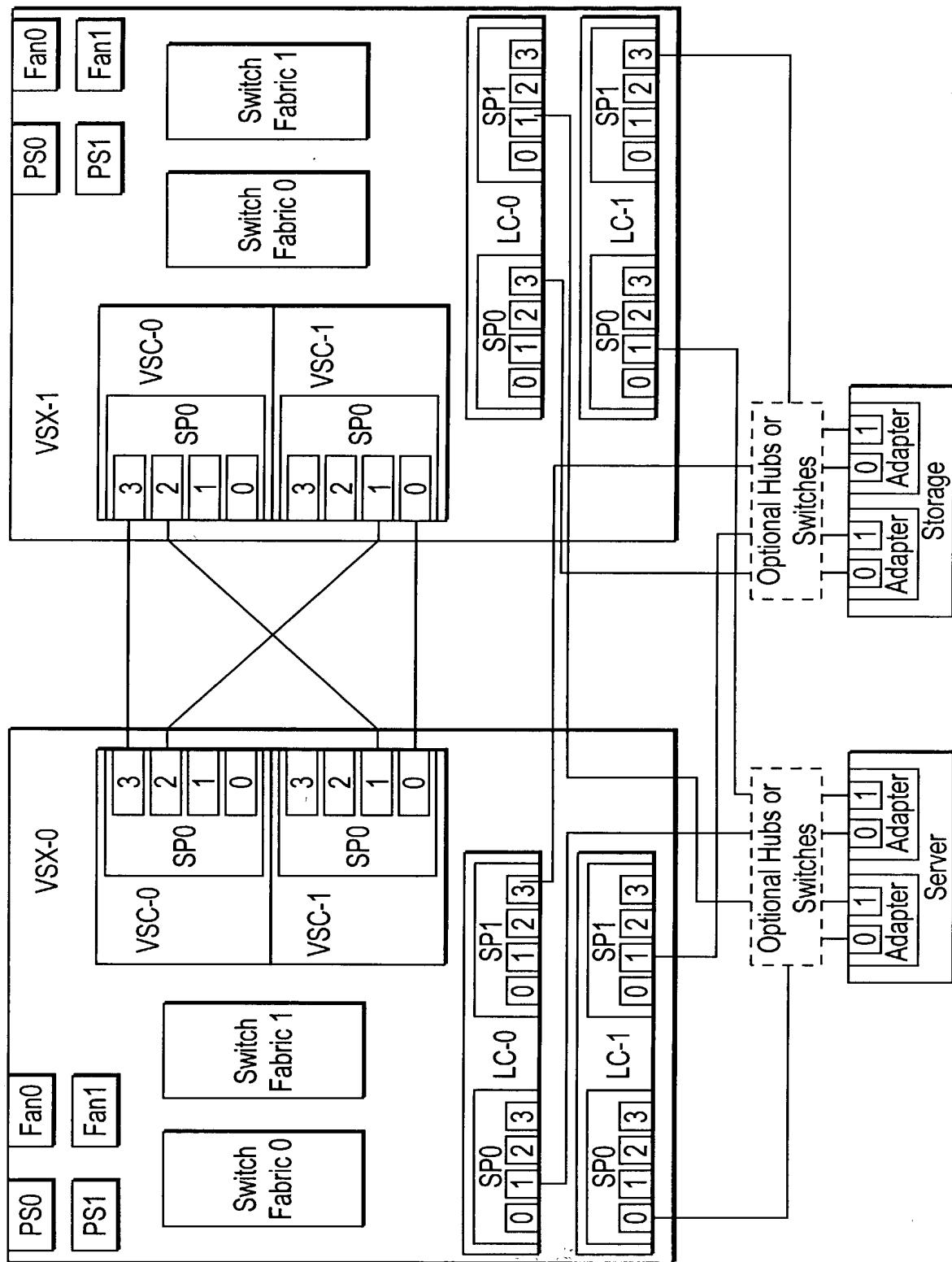
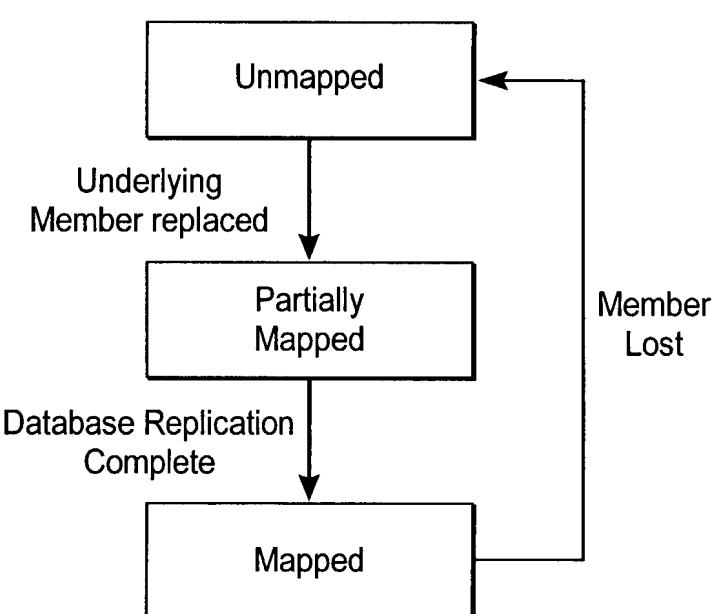
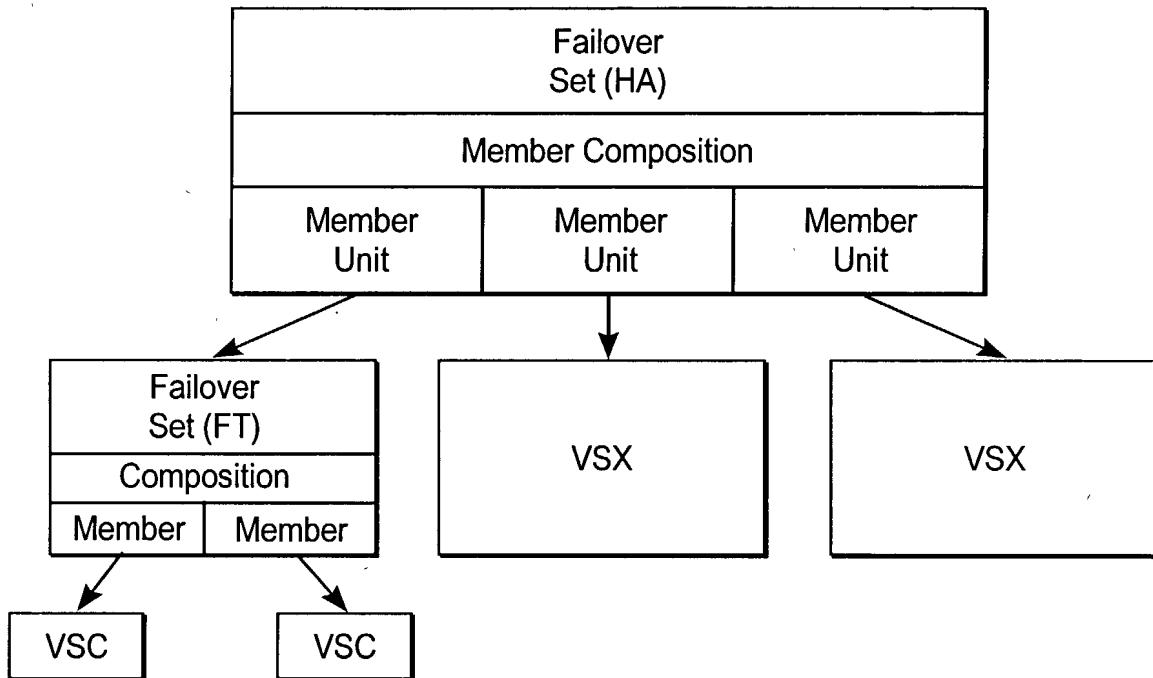


FIG. 4 High Availability Configuration



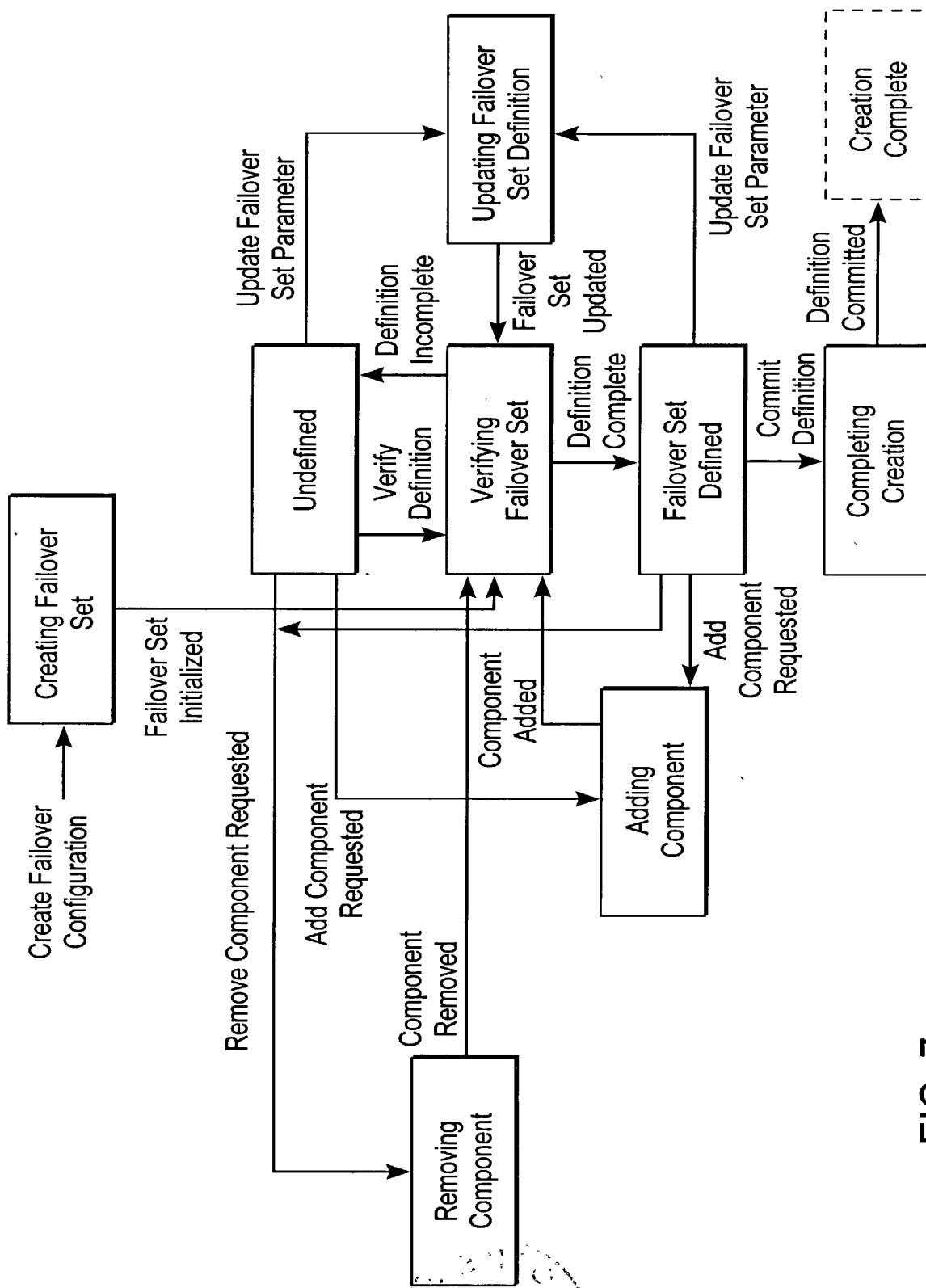


FIG. 7 Creating a Failover Set

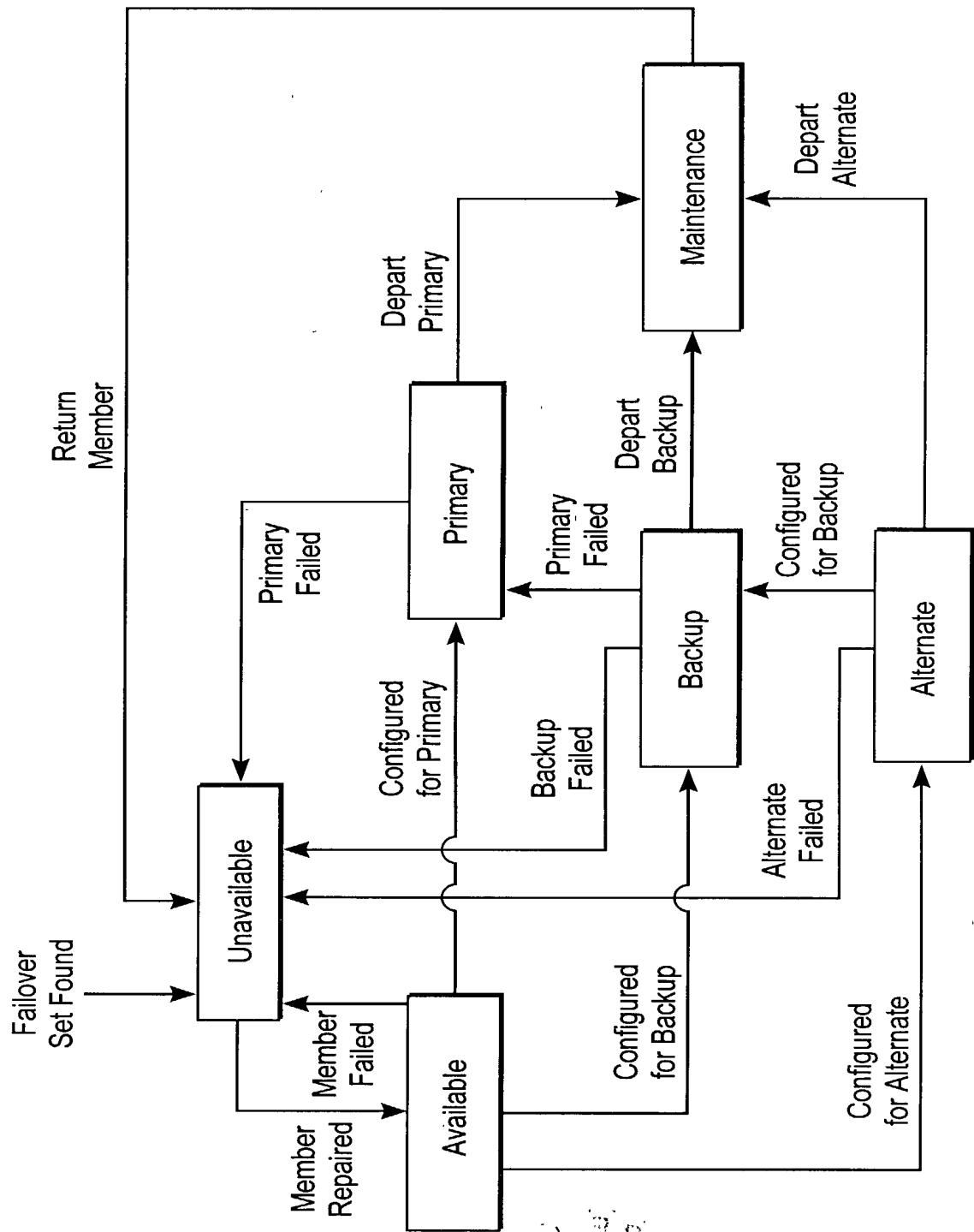


FIG. 8 Member State Diagram

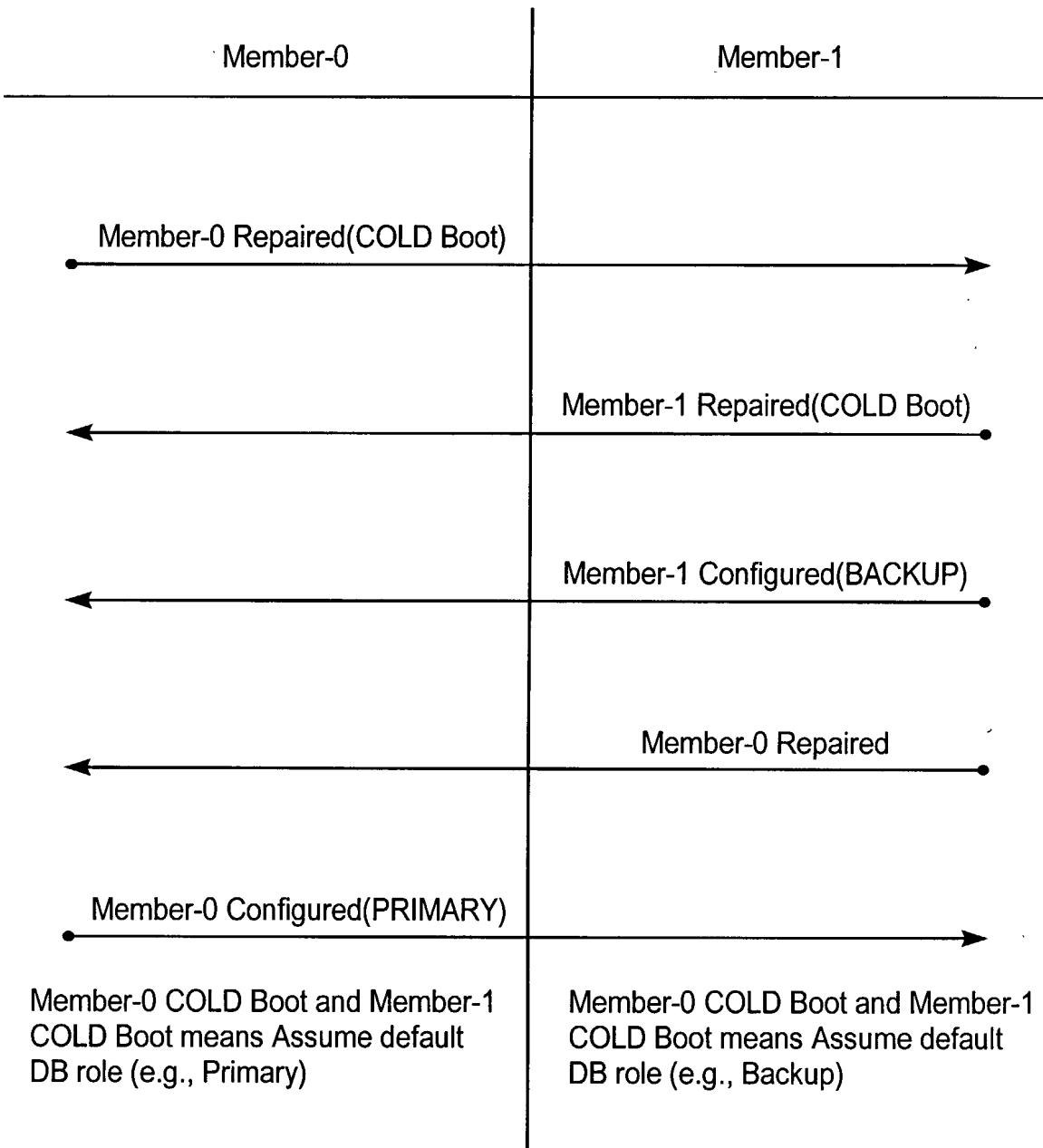


FIG. 9 Member Arbitration for COLD Boot

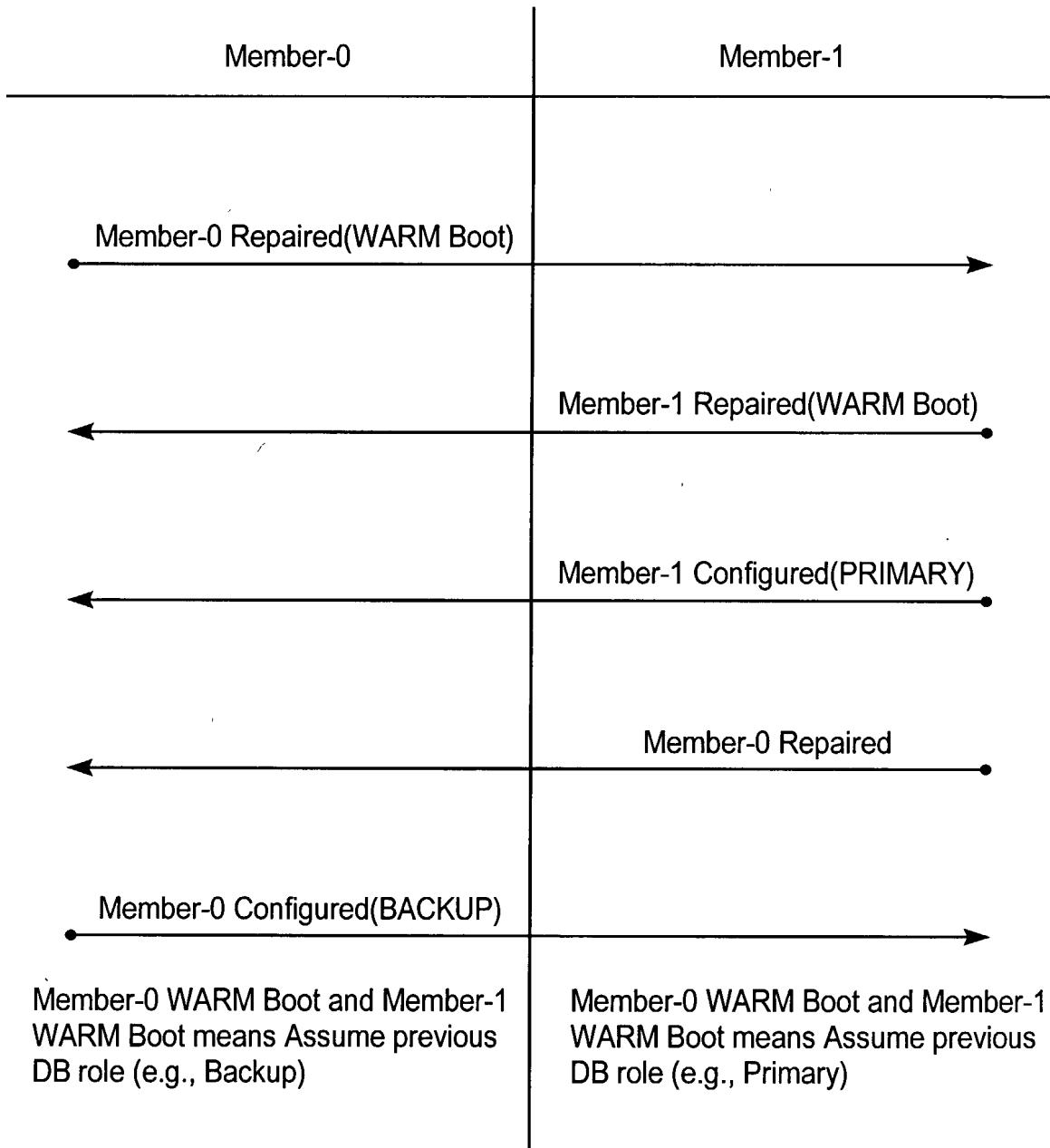


FIG. 10 Member Arbitration for WARM Boot

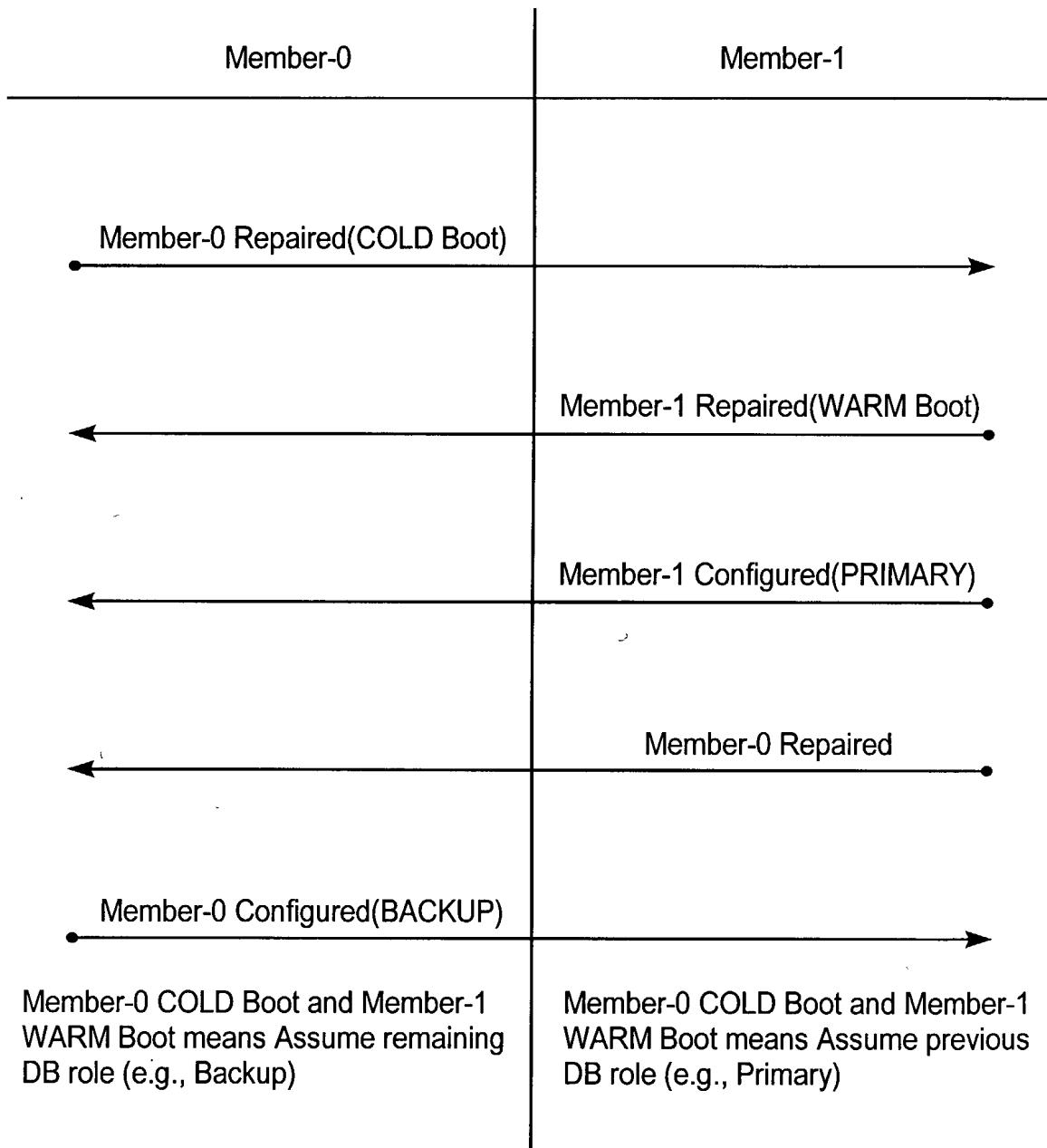


FIG. 11 Member Arbitration for Mixed Boot

Old State	Event	Mj Repaired	Mj Repaired	Mj Configured	Mj Failed	Mj Failed
1. $\{M_i, M_j\}$ Unavail, $\{ \}$ Avail, $\{ \}$ Primary, $\{ \}$ Backup*	New St: 3 Action: A	New St: 2 Action: B		New St: 1 Action: S	New St: 1 Action: T	
2. $\{M_i\}$ Unavail, $\{M_j\}$ Avail, $\{ \}$ Primary, $\{ \}$ Backup	New St: 4 Action: C		New St: 8 Action: D	New St: 2 Action: S		
3. $\{M_j\}$ Unavail, $\{M_i\}$ Avail, $\{ \}$ Primary, $\{ \}$ Backup		New St: 4 Action: E	New St: 9 Action: F		New St: 3 Action: T	
4. $\{ \}$ Unavail, $\{M_i, M_j\}$ Avail, $\{ \}$ Primary, $\{ \}$ Backup			New St: 7 Action: G	New St: 6 Action: H		
5a. $\{ \}$ Unavail, $\{ \}$ Avail, $\{M_i\}$ Primary, $\{M_j\}$ Backup					New St: 9 Action: I	
5b. $\{ \}$ Unavail, $\{ \}$ Avail, $\{M_j\}$ Primary, $\{M_i\}$ Backup					New St: 8 Action: J	
6. $\{ \}$ Unavail, $\{M_i\}$ Avail, $\{M_j\}$ Pri, $\{ \}$ Backup	New St: 6 Action: K		New St: 5a,5b Action: L		New St: 8 Action: I	
7. $\{ \}$ Unavail, $\{M_j\}$ Avail, $\{M_i\}$ Pri, $\{ \}$ Backup		New St: 7 Action: N		New St: 5a,5b Action: O	New St: 2 Action: P	
8. $\{M_i\}$ Unavail, $\{ \}$ Avail, $\{M_j\}$ Pri, $\{ \}$ Backup	New St: 6 Action: C					New St: 1 Action: Q
9. $\{M_j\}$ Unavail, $\{ \}$ Avail, $\{M_i\}$ Pri, $\{ \}$ Backup		New St: 7 Action: E			New St: 1 Action: R	
* Initial State						

FIG. 12 2 Member State Table

Action Routines	Description
1	1. Send "Mi repaired" to Mj, if Mj is not failed. 2. Set timer to send "Mi repaired" to Mi
2	1. Send "Mj repaired" to Mi, if Mi is not failed. 2. Set timer to send "Mj repaired" to Mj
A	1. If Mi and configured send "Mi configured" to Mj. 2. Set timer to send "Mj configured" to Mi. 3.
B	1. If Mj and configured send "Mj configured" to Mi. 2. Set timer to send "Mj configured" to Mj. 3.
C	1. If Mj, echo event back to Mi. 2. If Mi and configured send "Mi configured" to Mj. 3. Set timer to
D	1. If Mj, become Primary. 2. Otherwise, nop.
E	1. If Mi, echo event back to Mj. 2. If Mj and configured send "Mj configured" to Mi. 3. Set timer to
F	1. If Mi, become Primary. 2. Otherwise, nop.
G	1. If Mi, become Primary. 2. Otherwise, echo event back to Mi.
H	1. If Mj, become Primary. 2. Otherwise, echo event back to Mi.
I	1. If Mj, become Primary. 2. If Mi become Backup.
J	1. If Mi, become Primary. 2. If Mj become Backup.
K	1. If Mj, echo event back to Mi. 2. Otherwise, nop
L	1. If Mj, determine Member Role. 2. SEnd "Mi configured" to Mi when done. 3. If Mi determine
M	1. If Mj, perform Fail-Stop processing. 2. Send "Mj Failed" to Mi. 3. Otherwise become Primary after
N	1. If Mi, echo event back to Mj. 2. Otherwise, nop
O	1. If Mi, determine Member role. 2. Send "Mj configured" to Mj when done. 3. If Mi determine
P	1. If Mi, perform Fail-Stop processing. 2. Send "Mi Failed" to Mj. 3. Otherwise become Primary after
Q	1. If Mj, perform Fail-Stop processing for Mi. 2. Otherwise nop.
R	1. If Mi, perform Fail-Stop processing for Mi. 2. Otherwise nop.
S	1. Perform Fail-Stop processing for Mi
T	1. Perform Fail-Stop processing for Mj

FIG. 13 Action Routines for a 2 Node Configuration

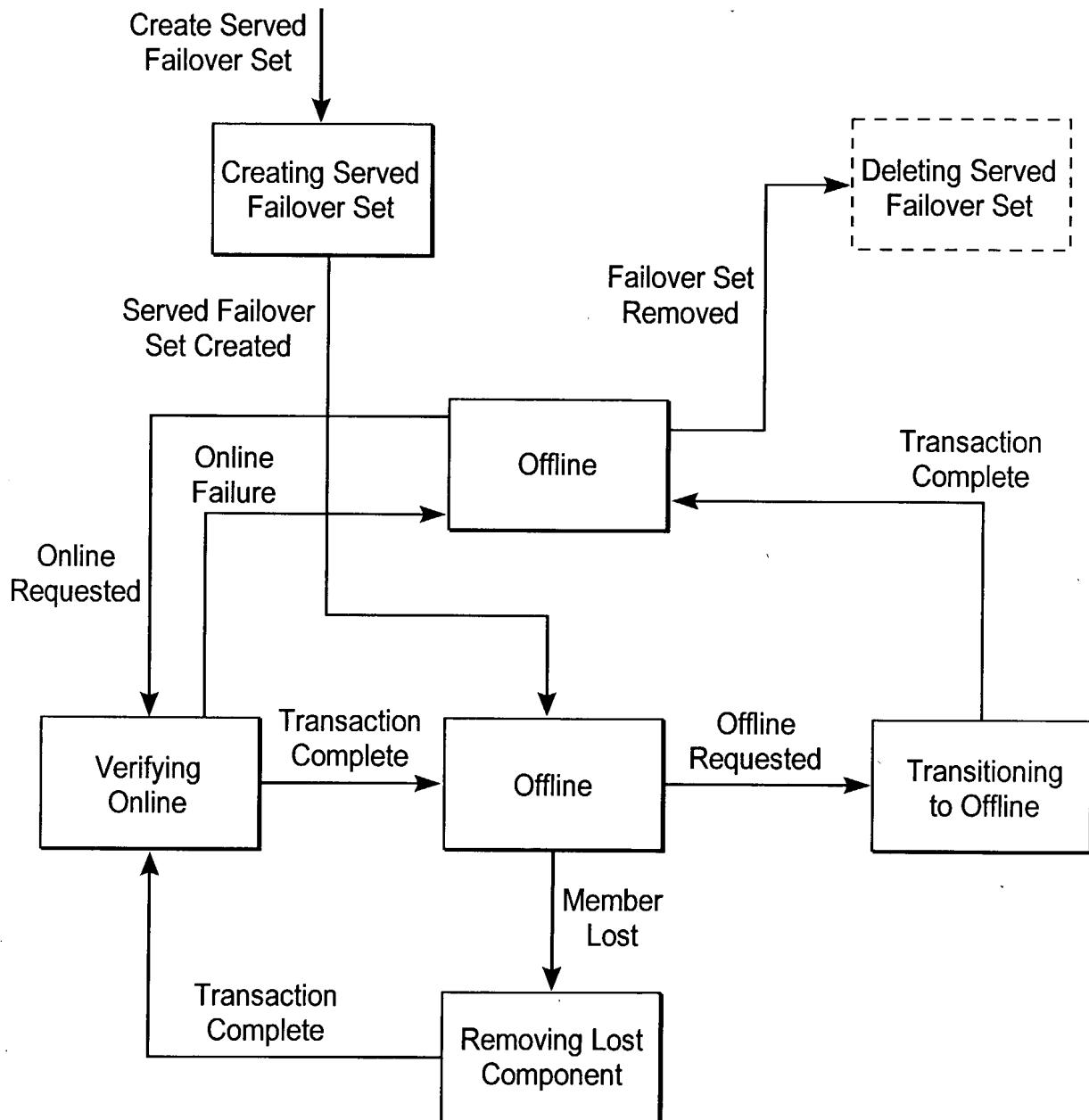


FIG. 14 Served Failover Set State Machine Diagram

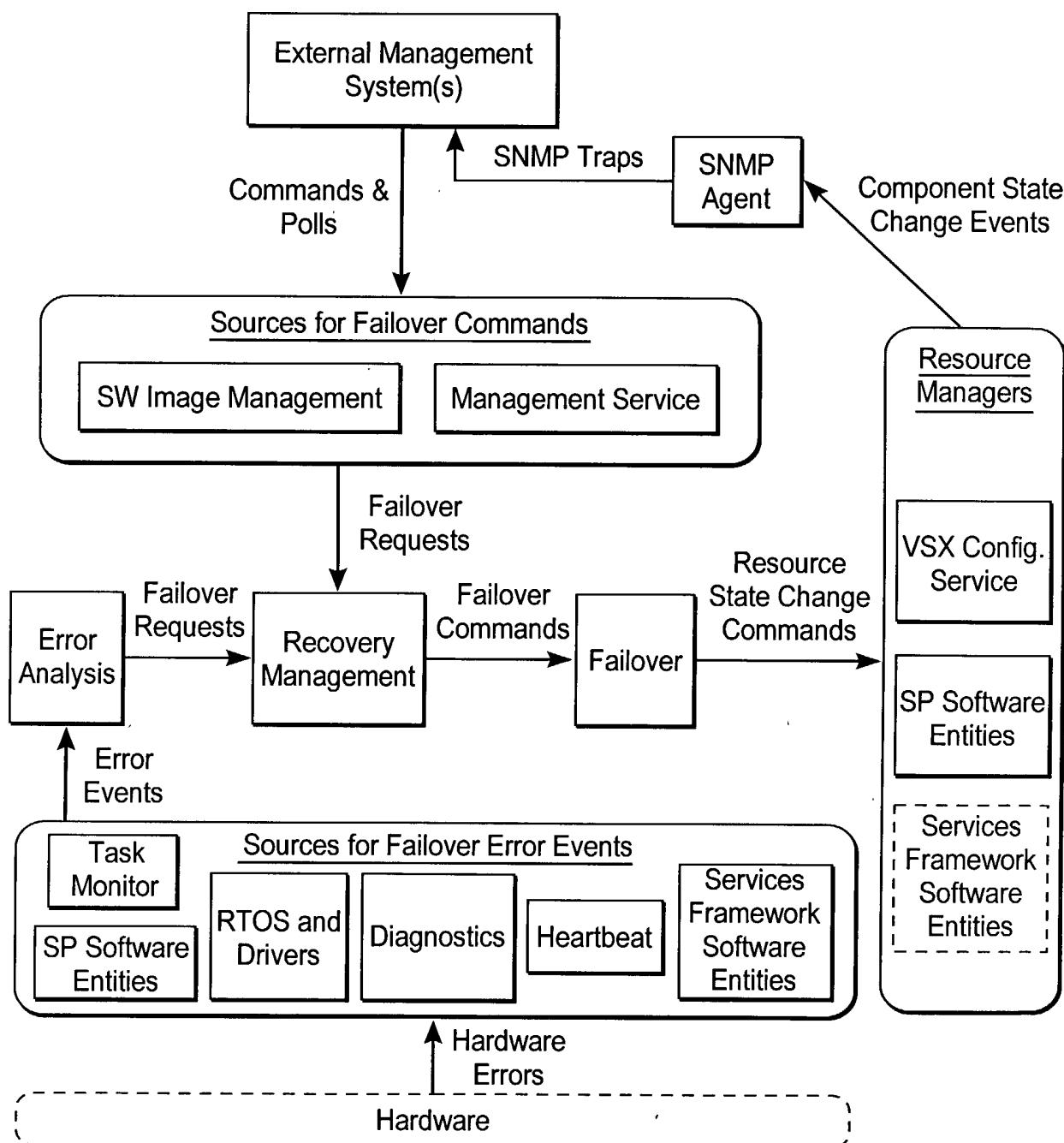
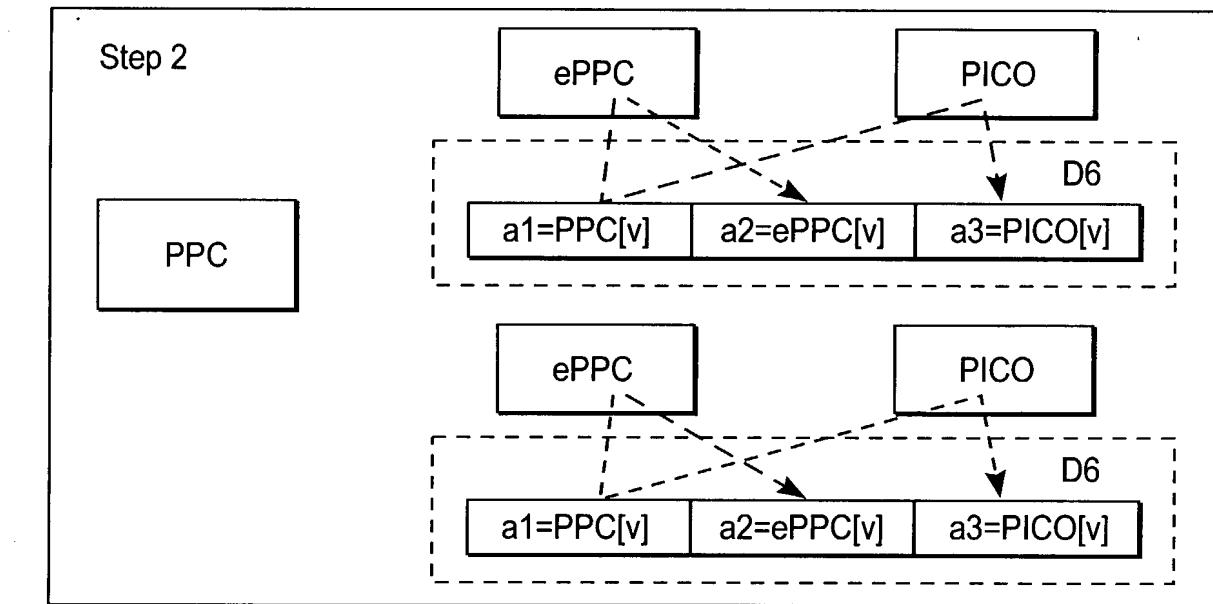
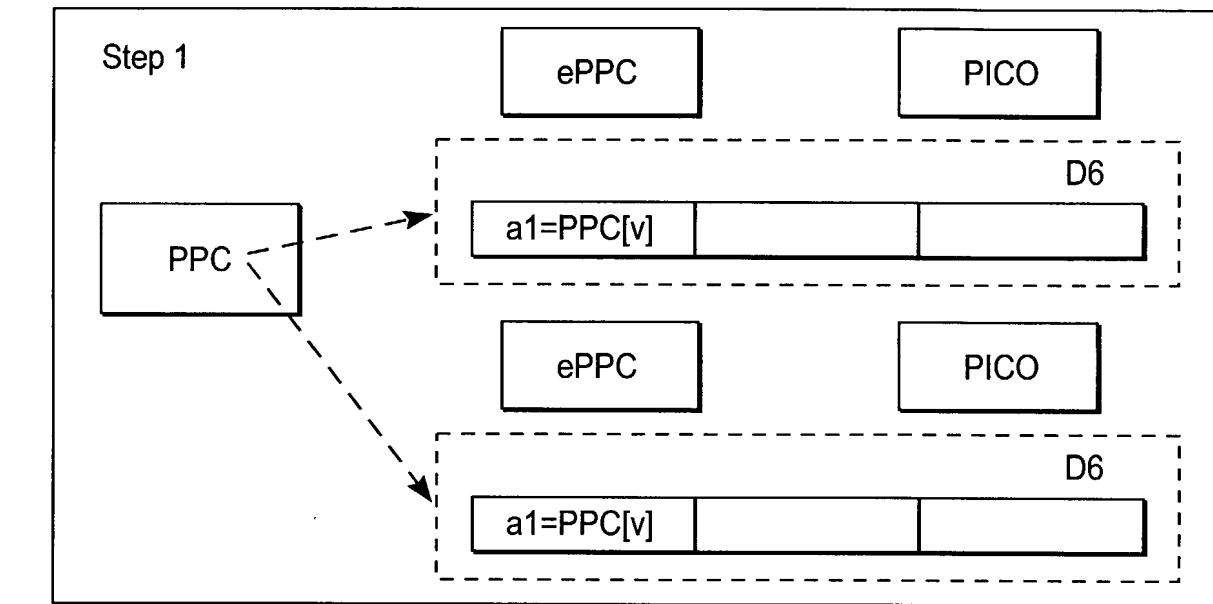


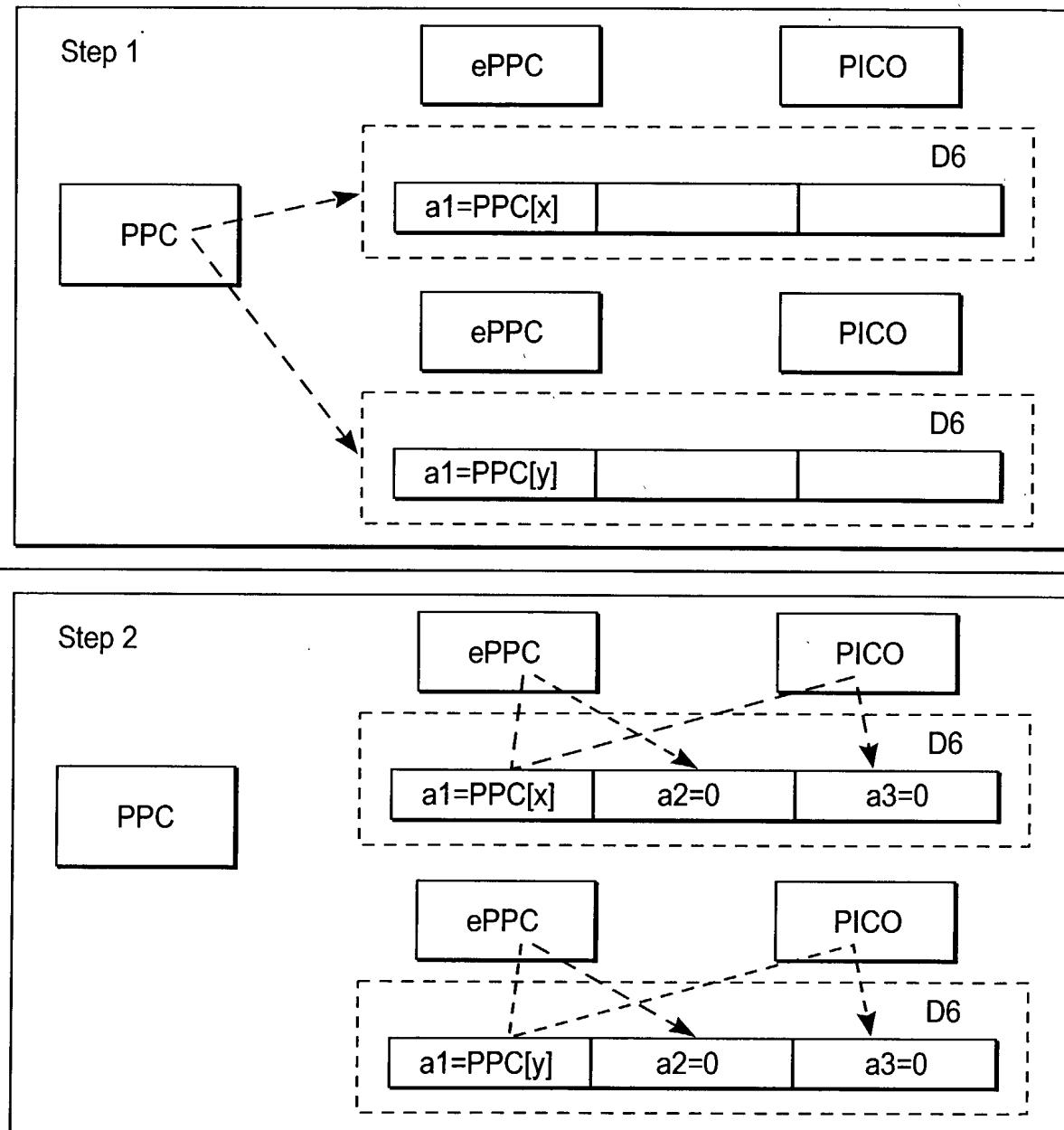
FIG. 15 Fault Detection and Analysis Architecture



Step 3

$\text{majority}(a1, a2, a3) = \text{majority}(v, v, v) = v$, No faults

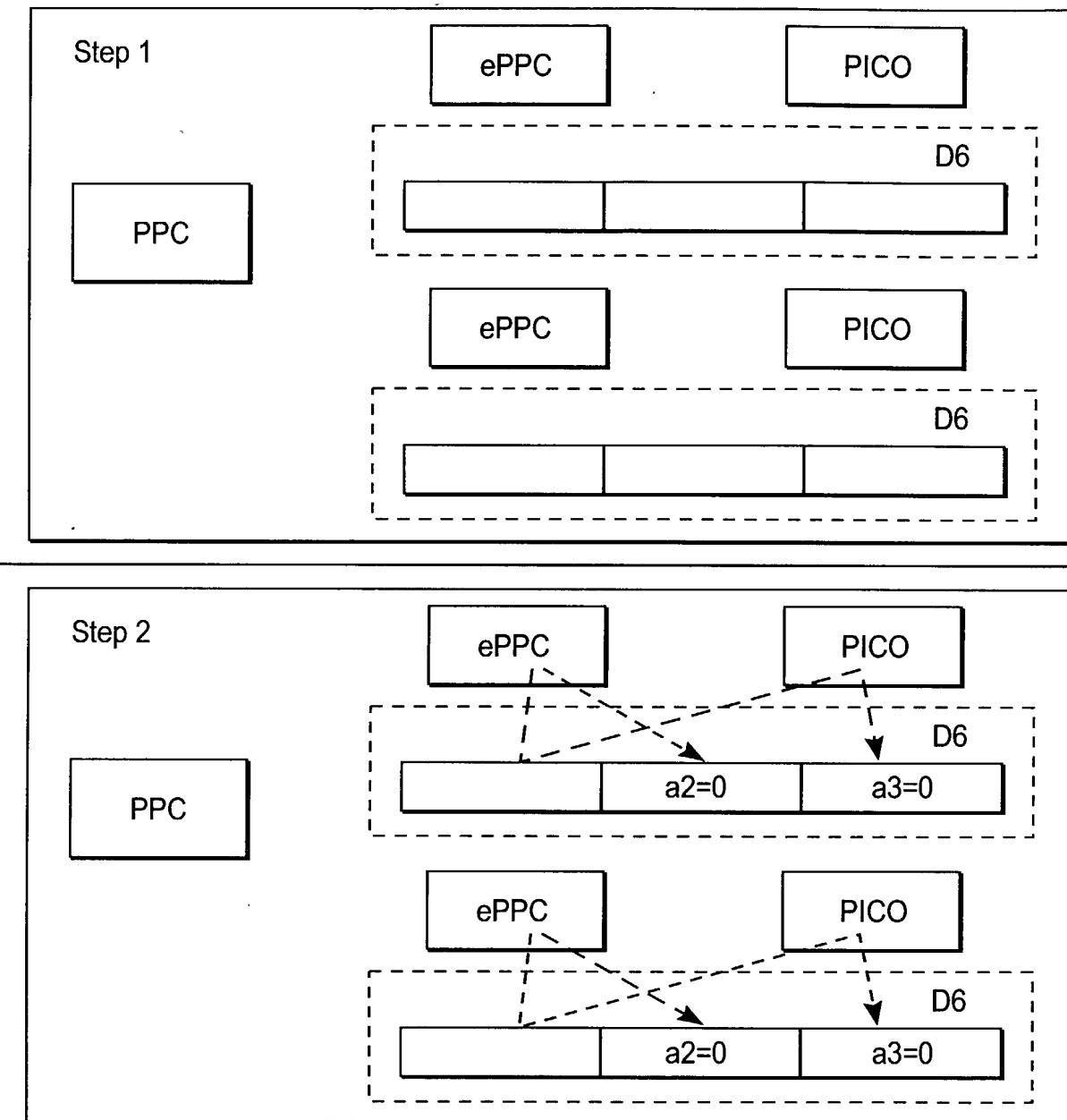
FIG. 16 No Faults



Step 3

$\text{majority}(a1, a2, a3) = \text{majority}(x, 0, 0) = 0$, transmitter fault

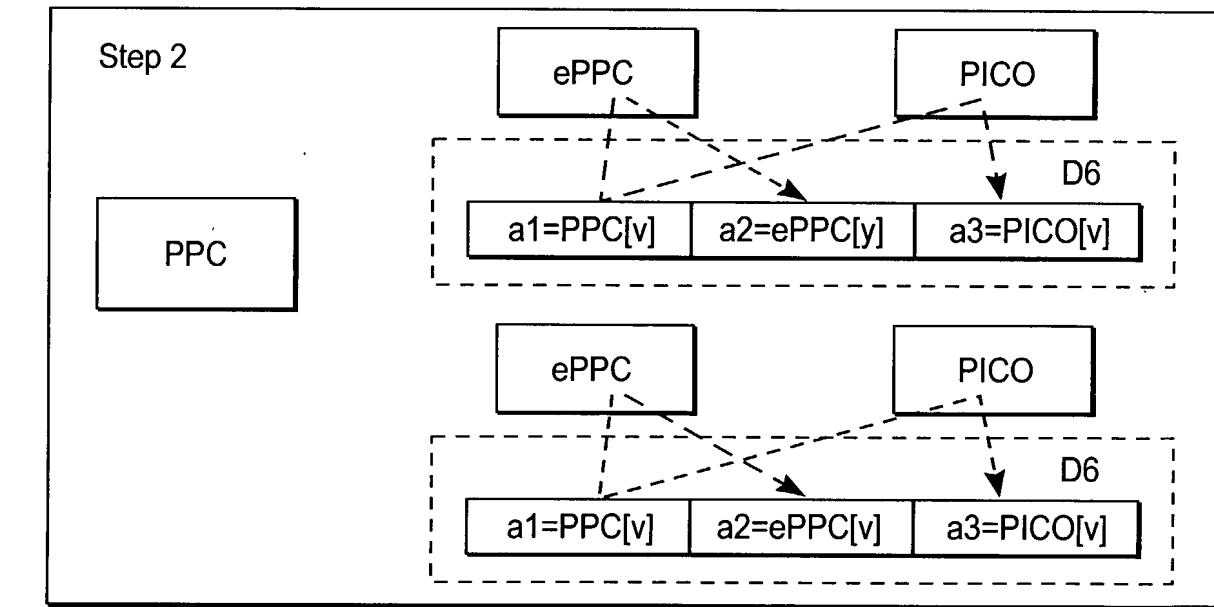
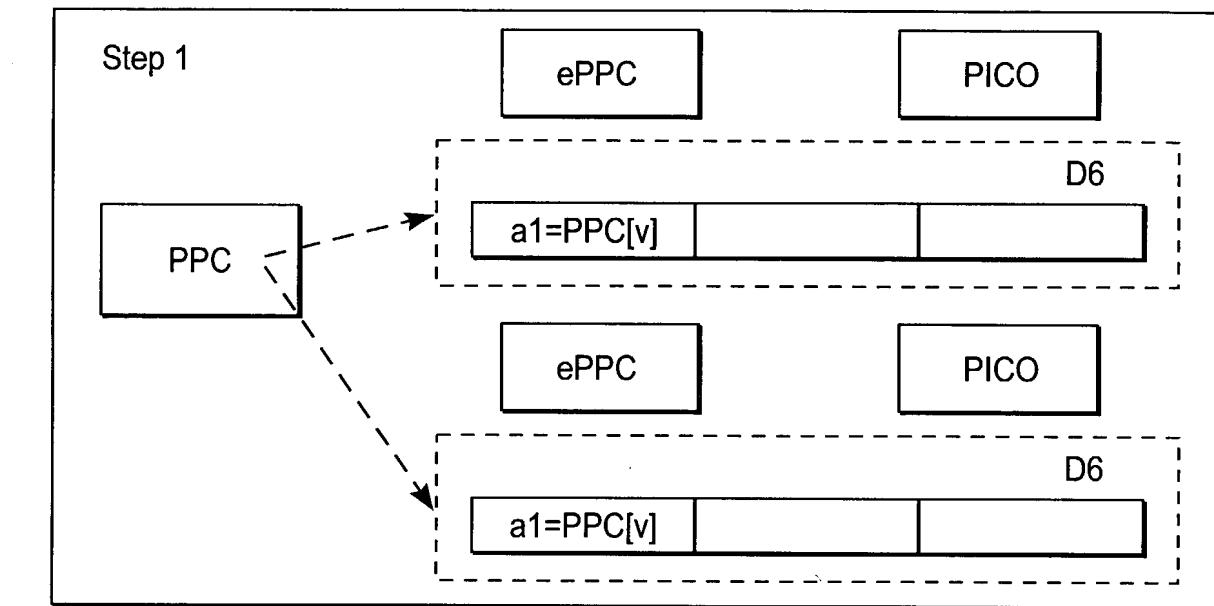
FIG. 17 Transmitter fault (sends a bad value)



Step 3

$\text{majority}(a1, a2, a3) = \text{majority}(0, 0, 0) = 0$, transmitter fault

FIG. 18 Transmitter fault (doesn't send a value)



Step 3

majority($a1, a2, a3$) = majority(v, y, v) = v , Receiver fault

FIG. 19 Receiver fault (relays wrong value)

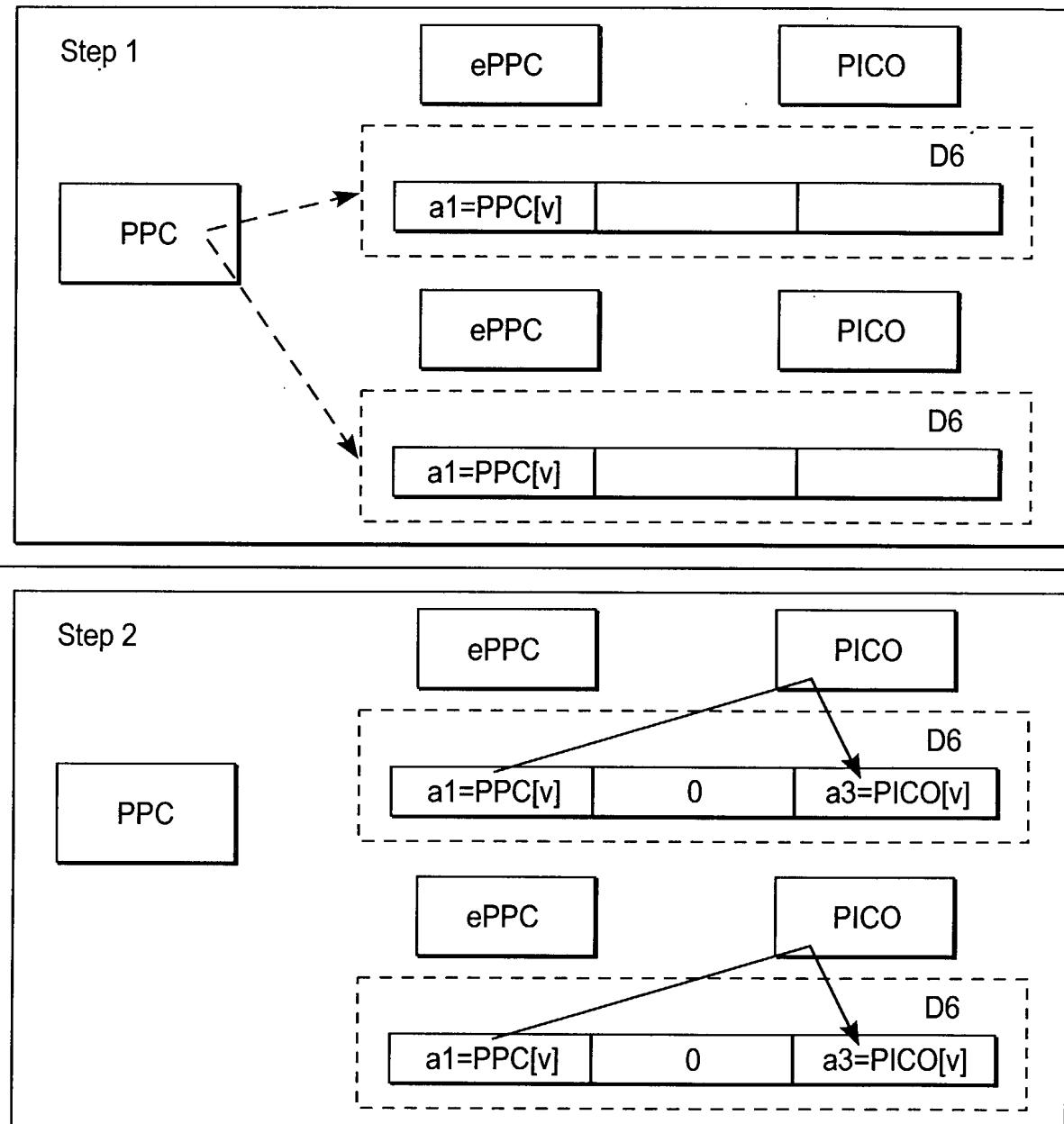


FIG. 20 Receiver fault (doesn't relay a value)

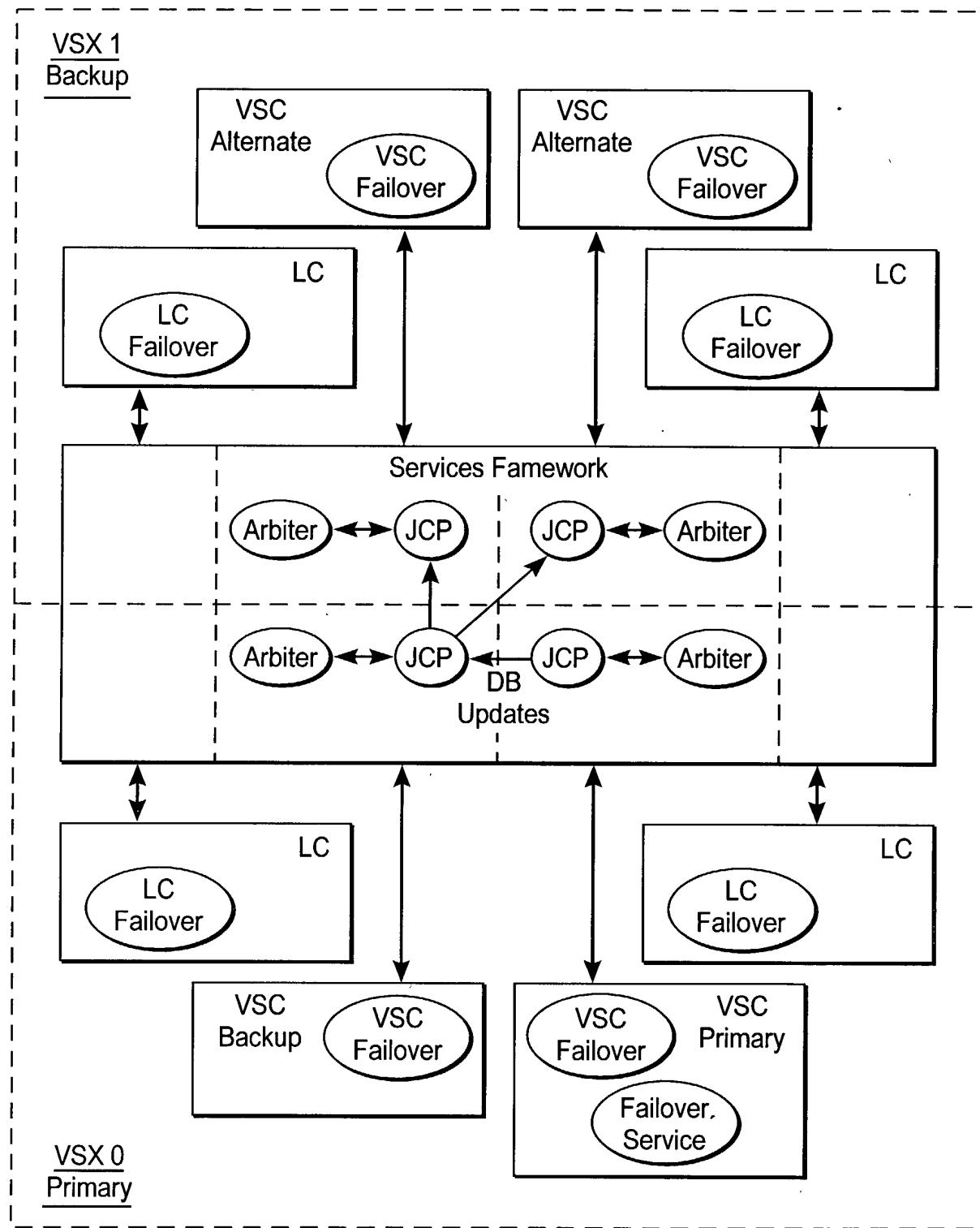


FIG. 21A Failover Service Architecture

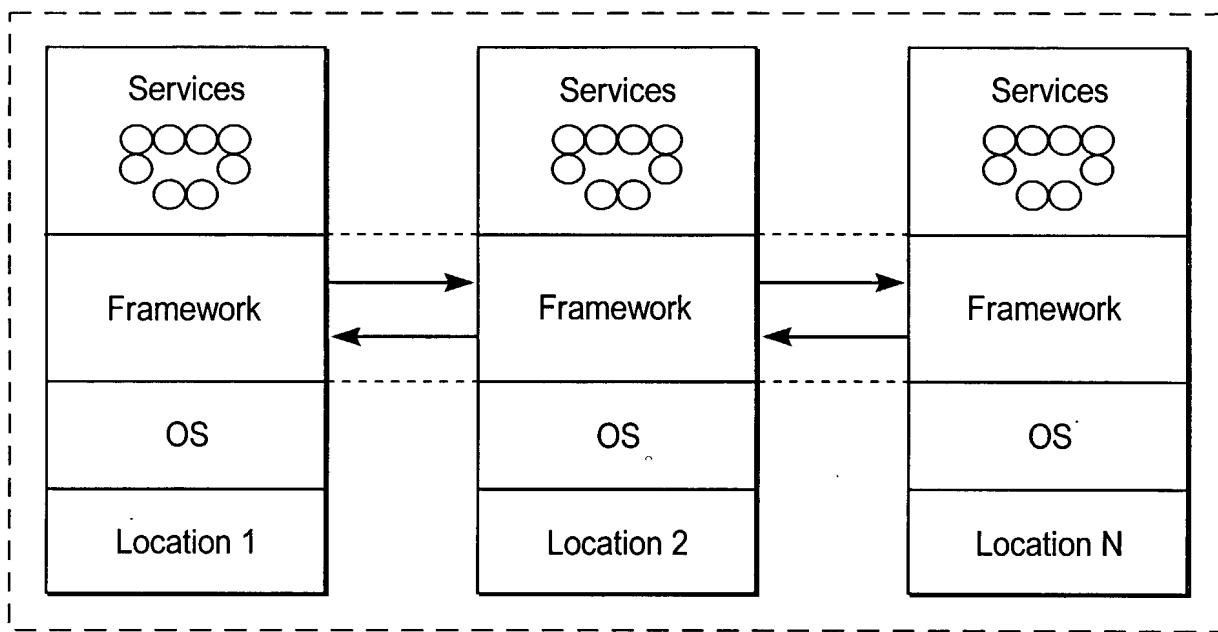


FIG. 21B

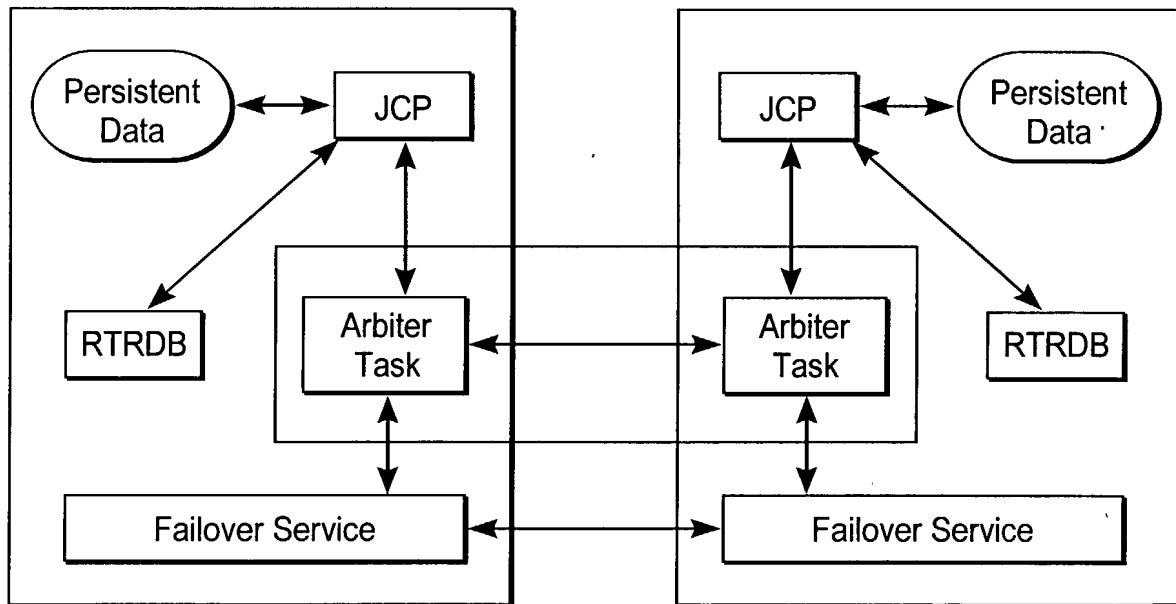


FIG. 22 An Arbiter for the Database

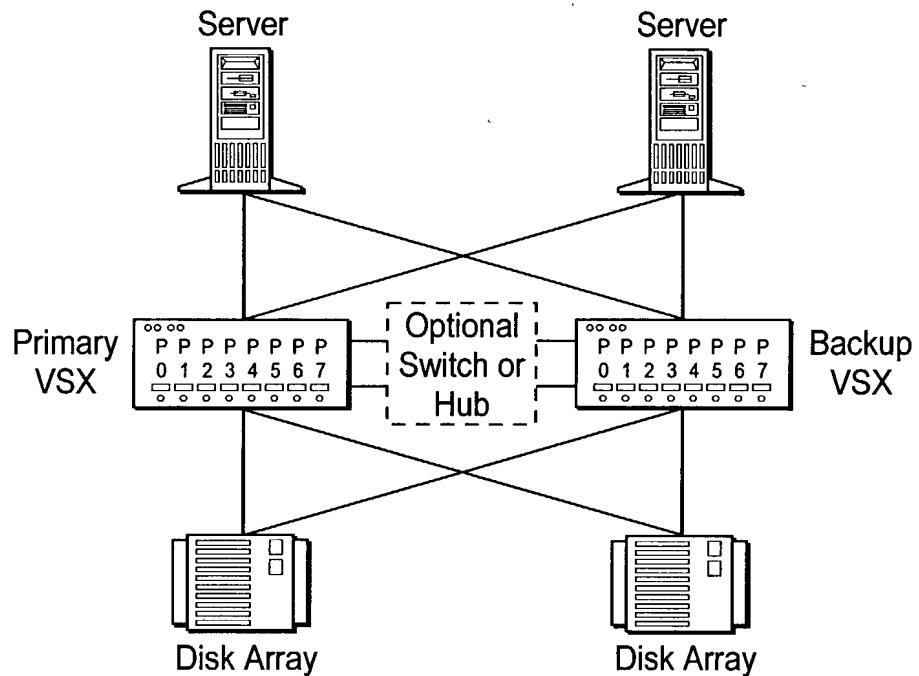


FIG. 23 Shared Link

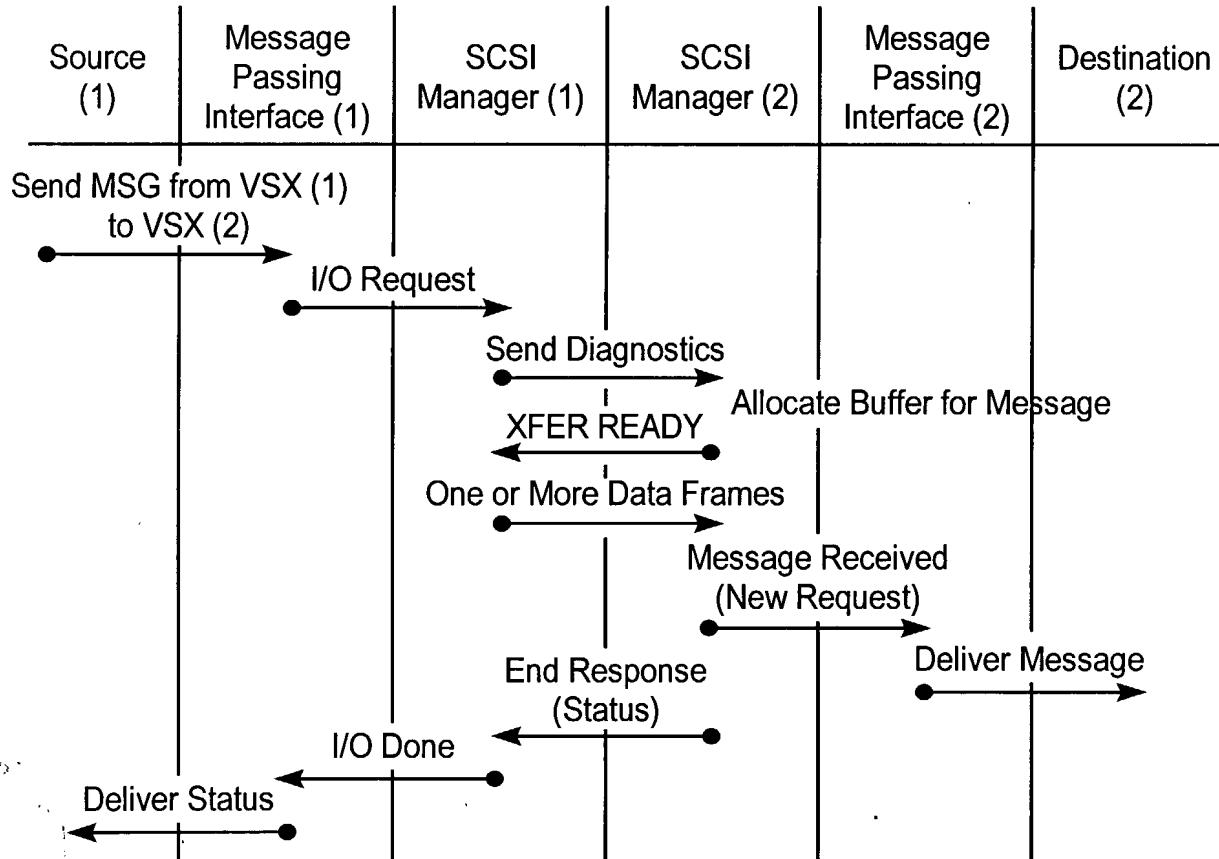


FIG. 24 VSX to VSX Message Passing

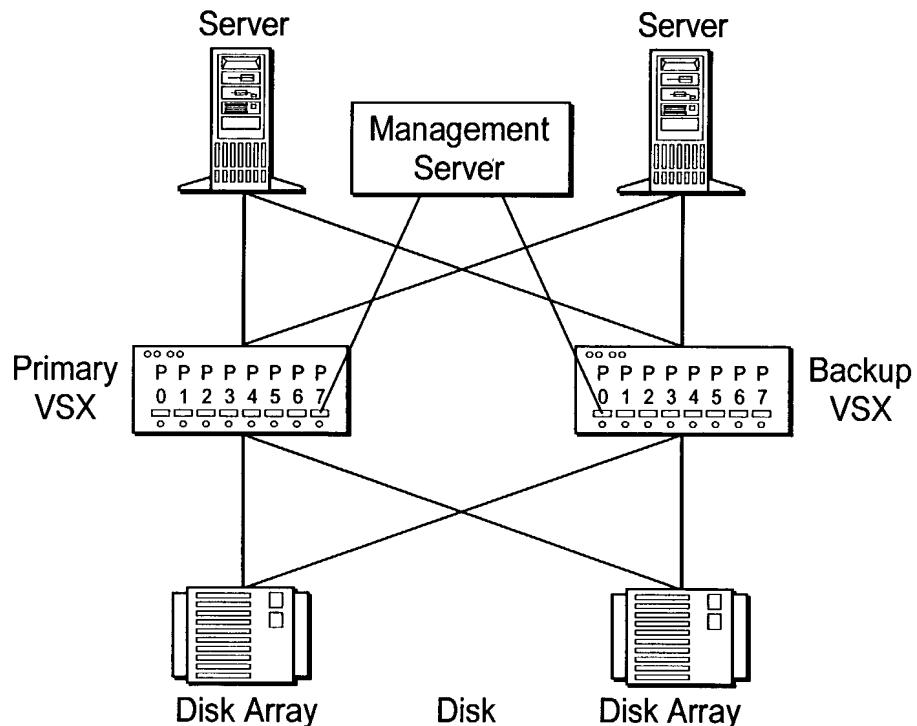


FIG. 25 Management Link

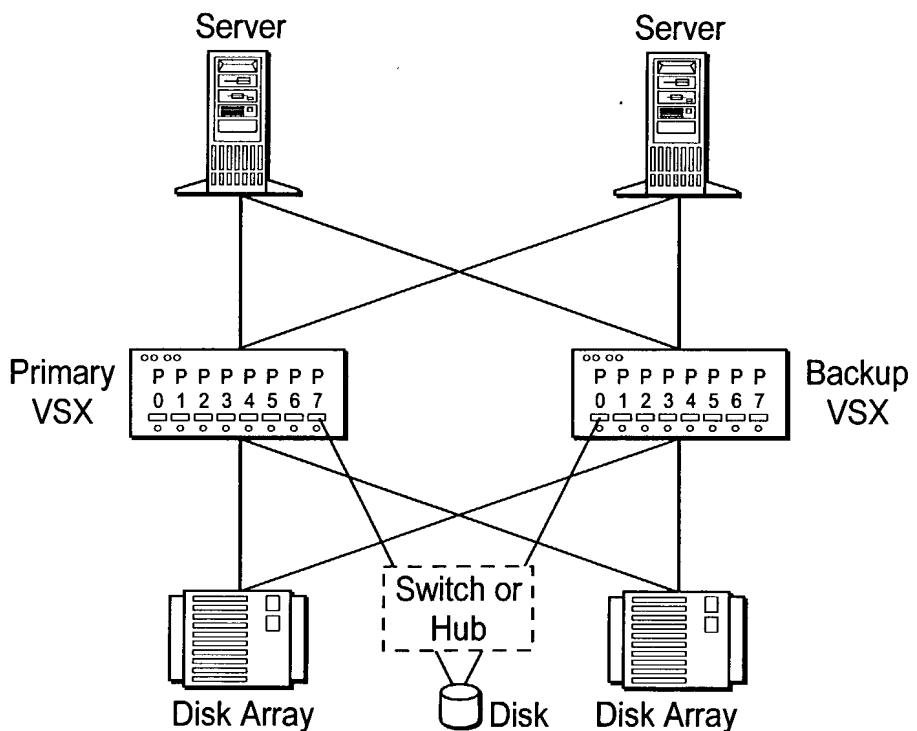


FIG. 26 Shared Disk

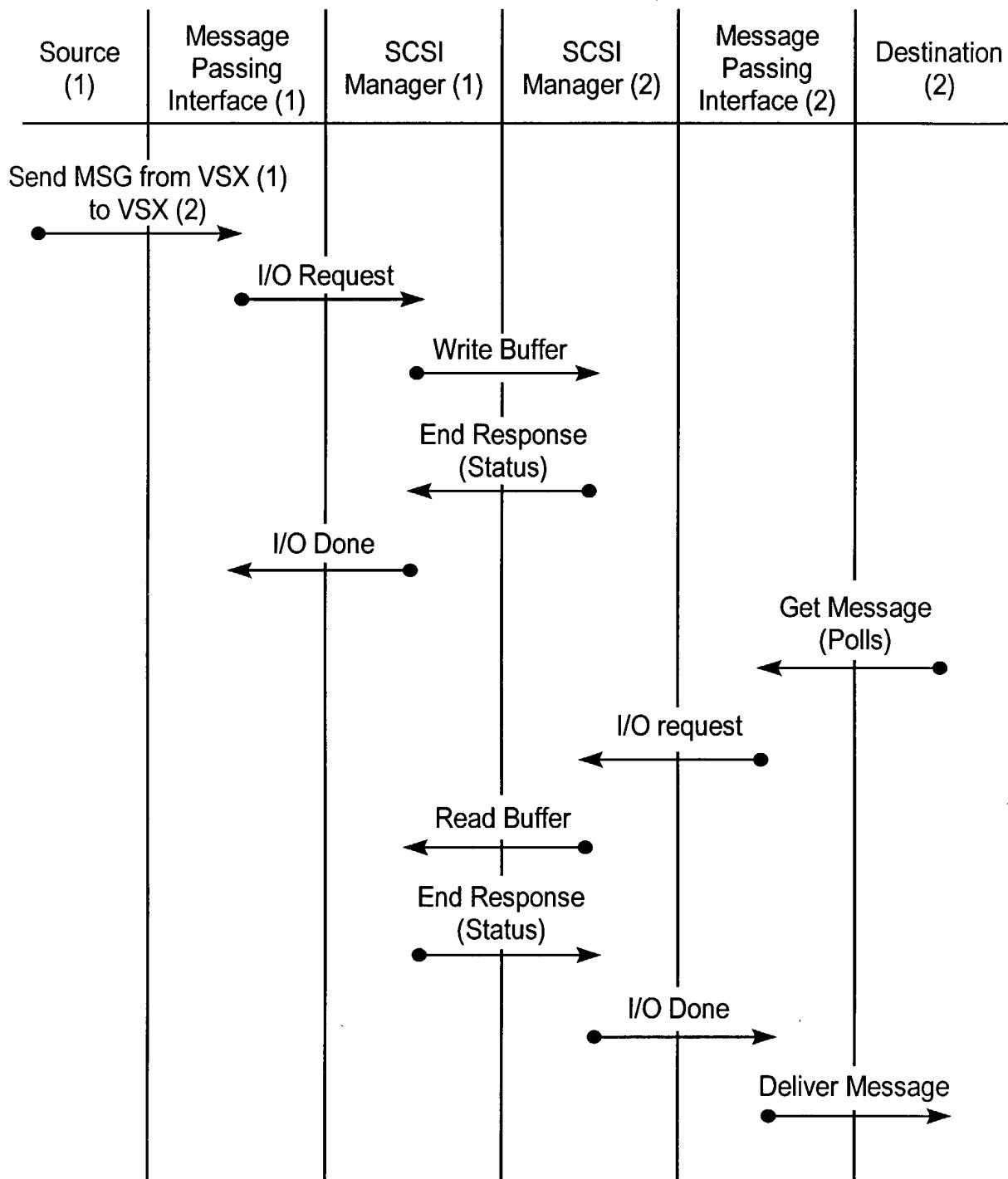


FIG. 27 VSX to VSX Communication Using Shared Disk

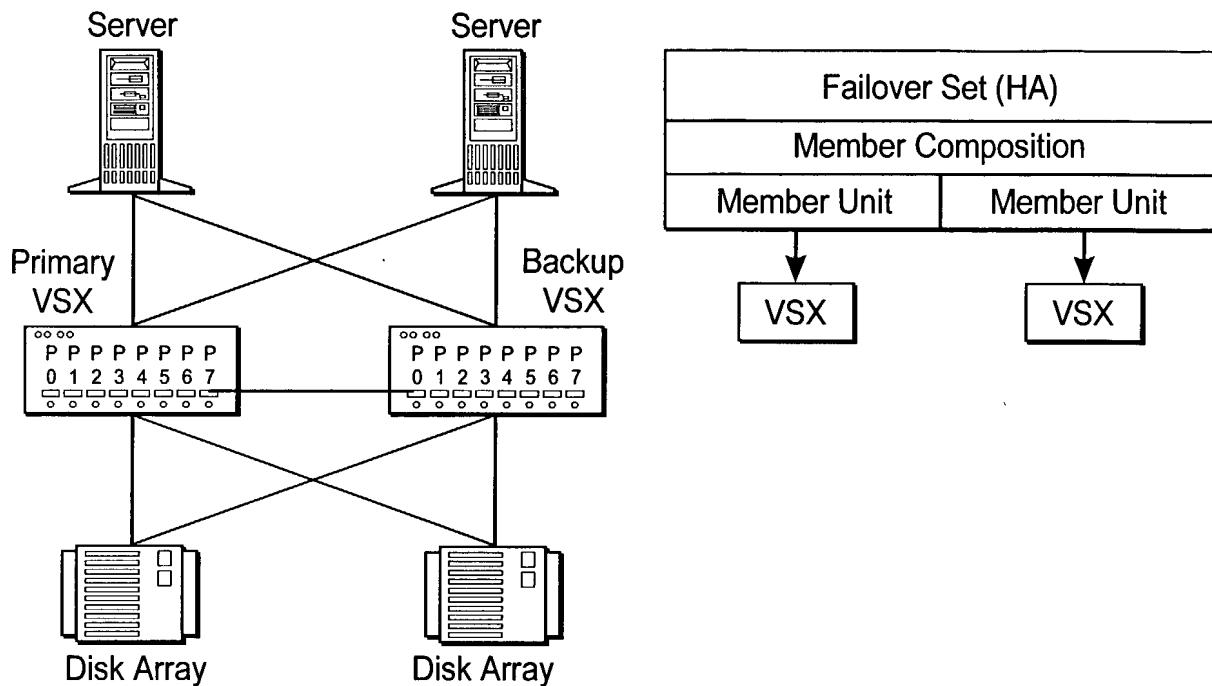


FIG. 28 2 Node HA Configuration

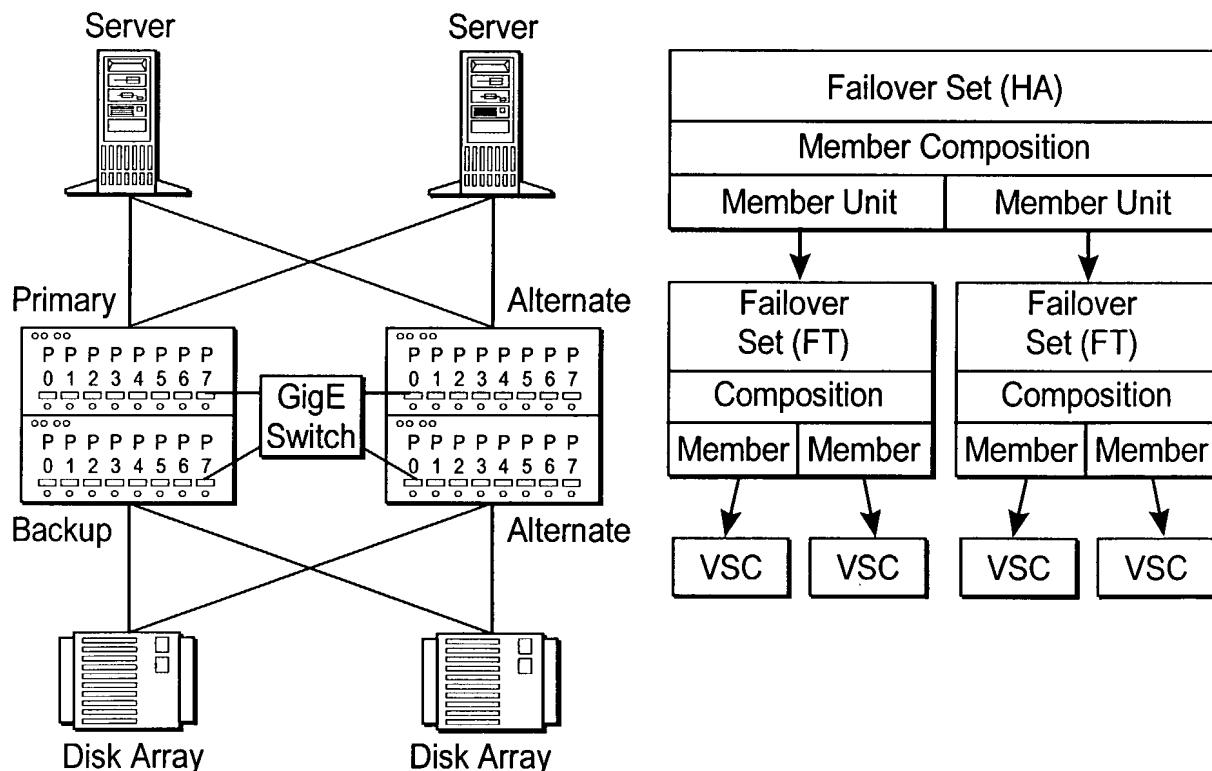


FIG. 29 Hierarchical HA Configuration

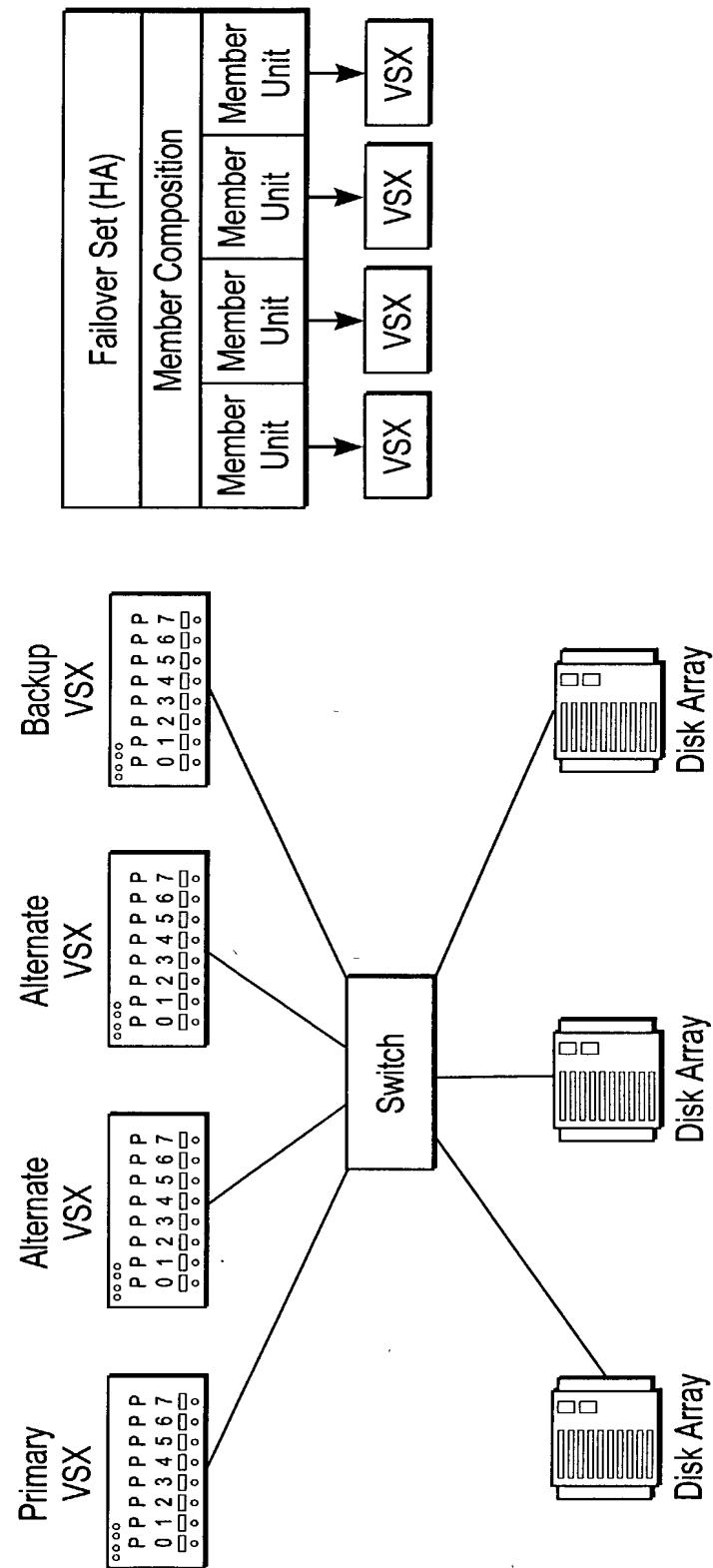


FIG. 30 N + 1 Nodes

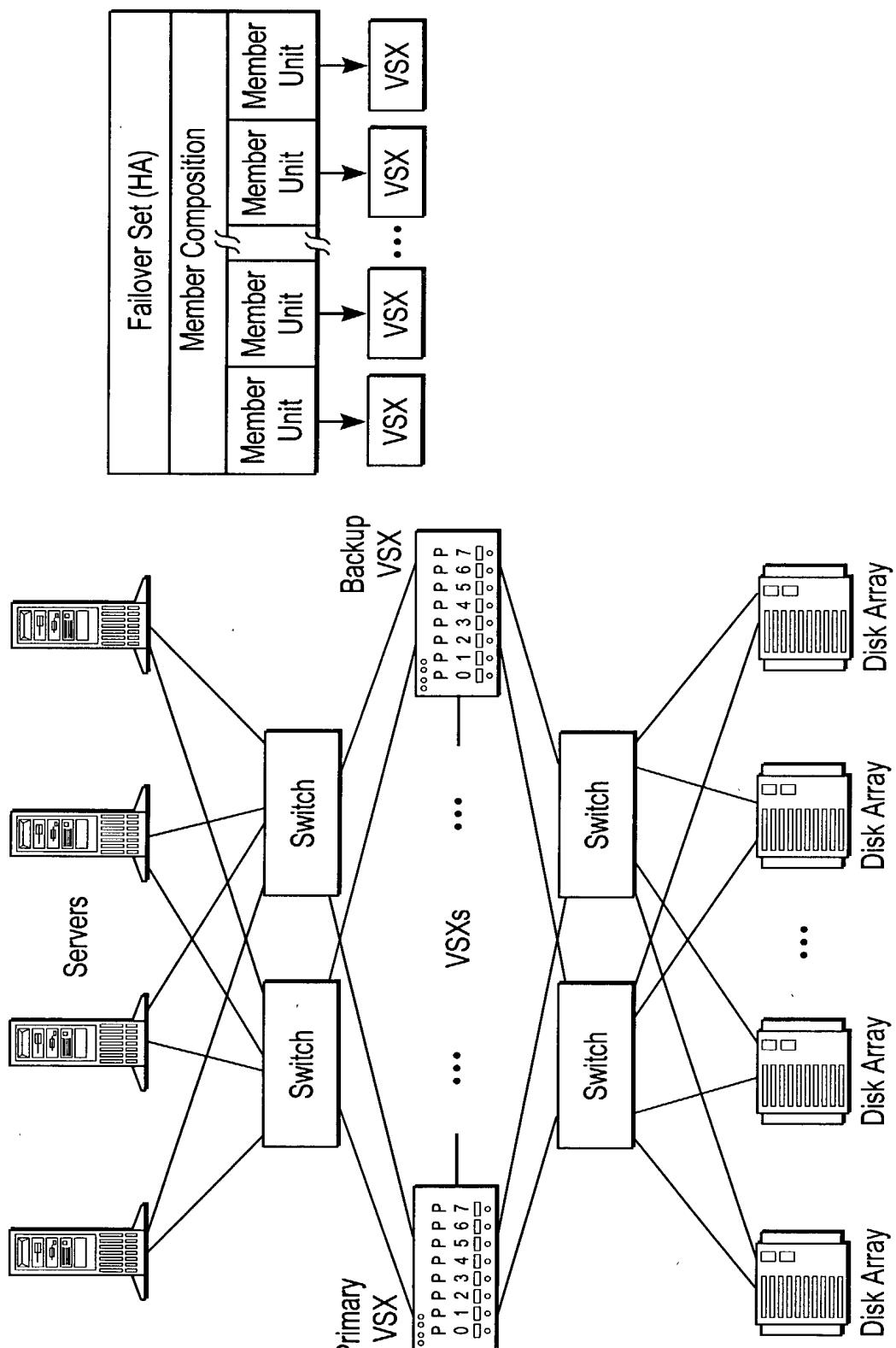


FIG. 31 N - Nodes

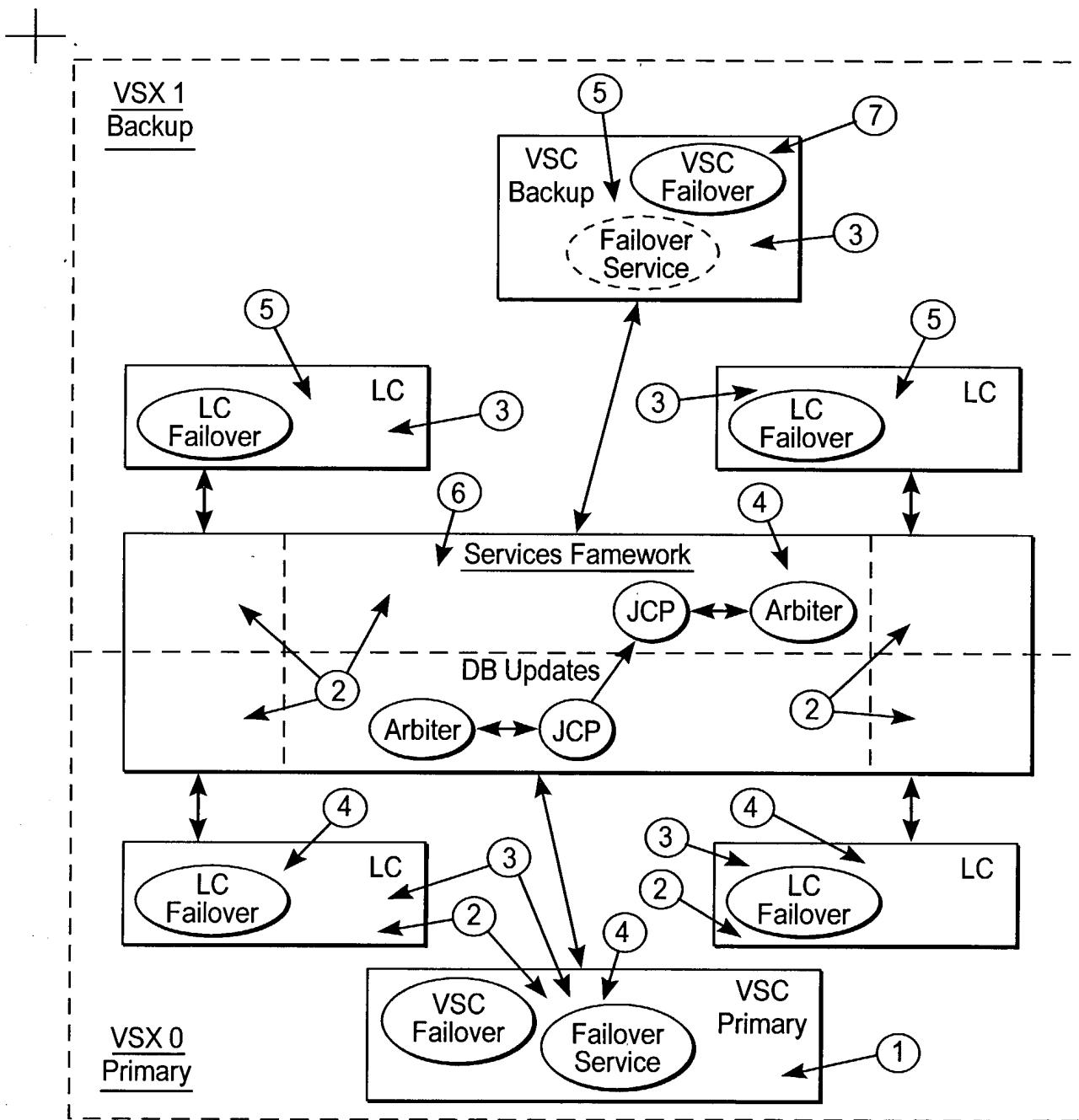


FIG. 32 VSX Failover, Primary Fails

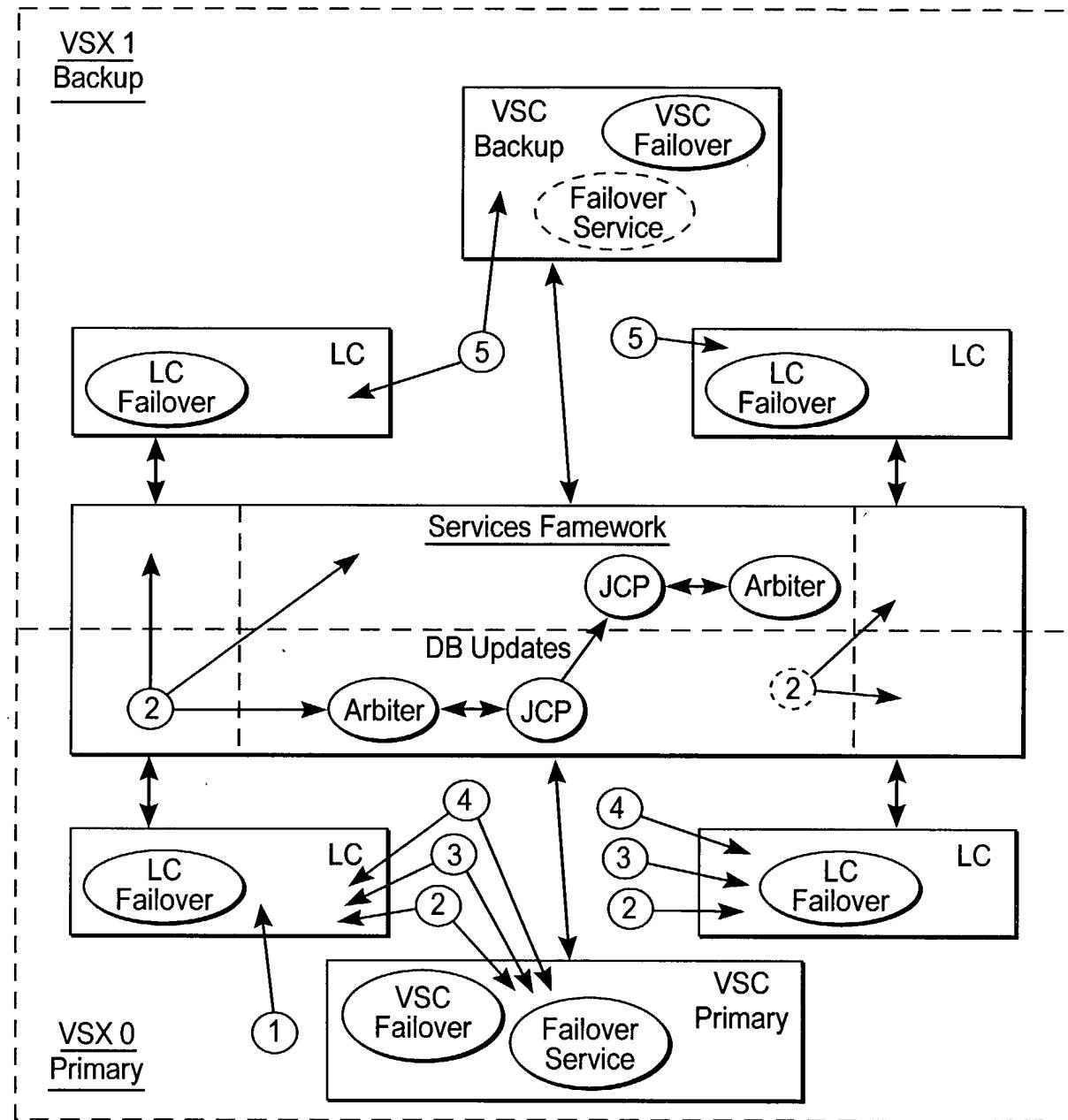
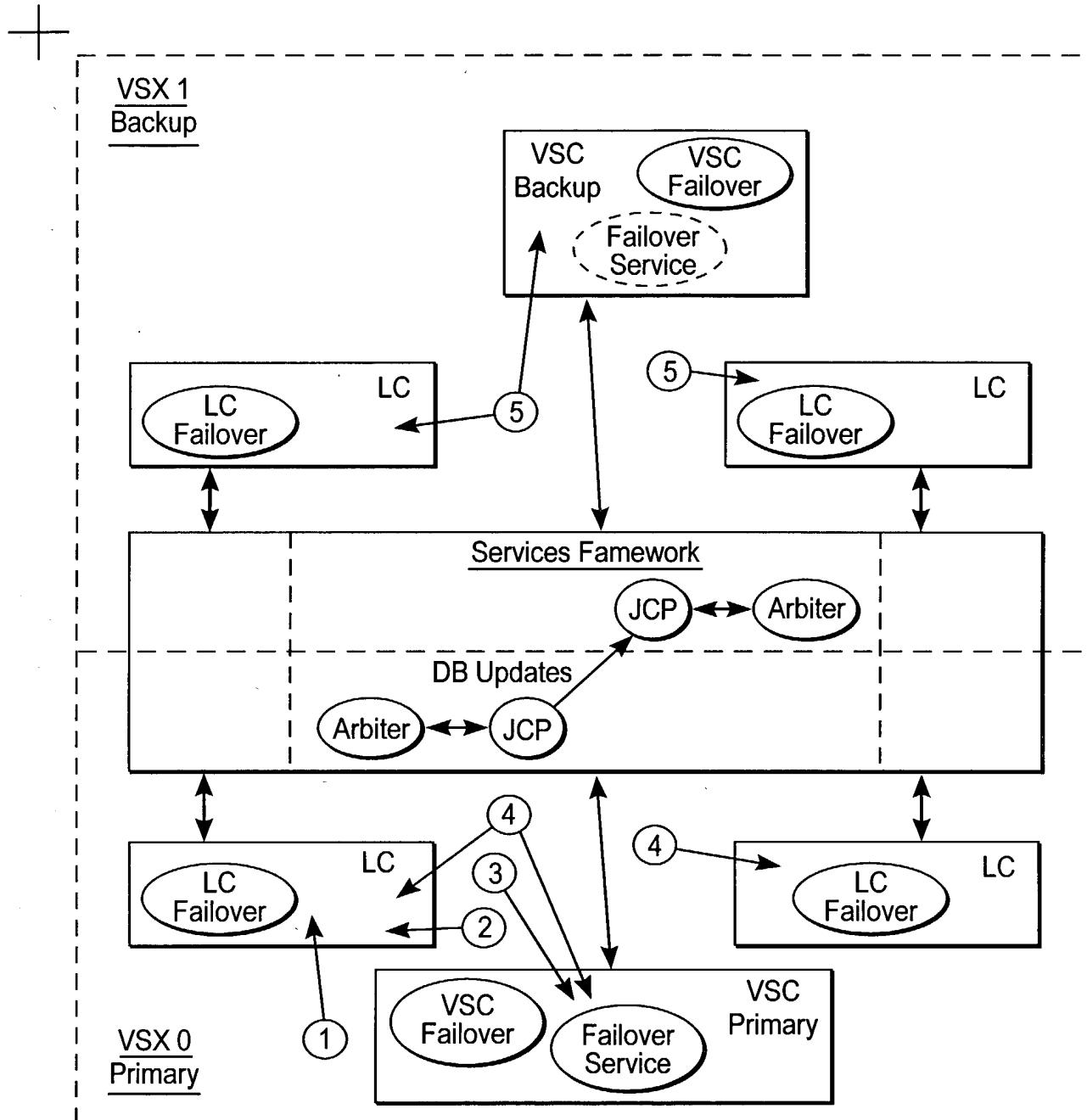
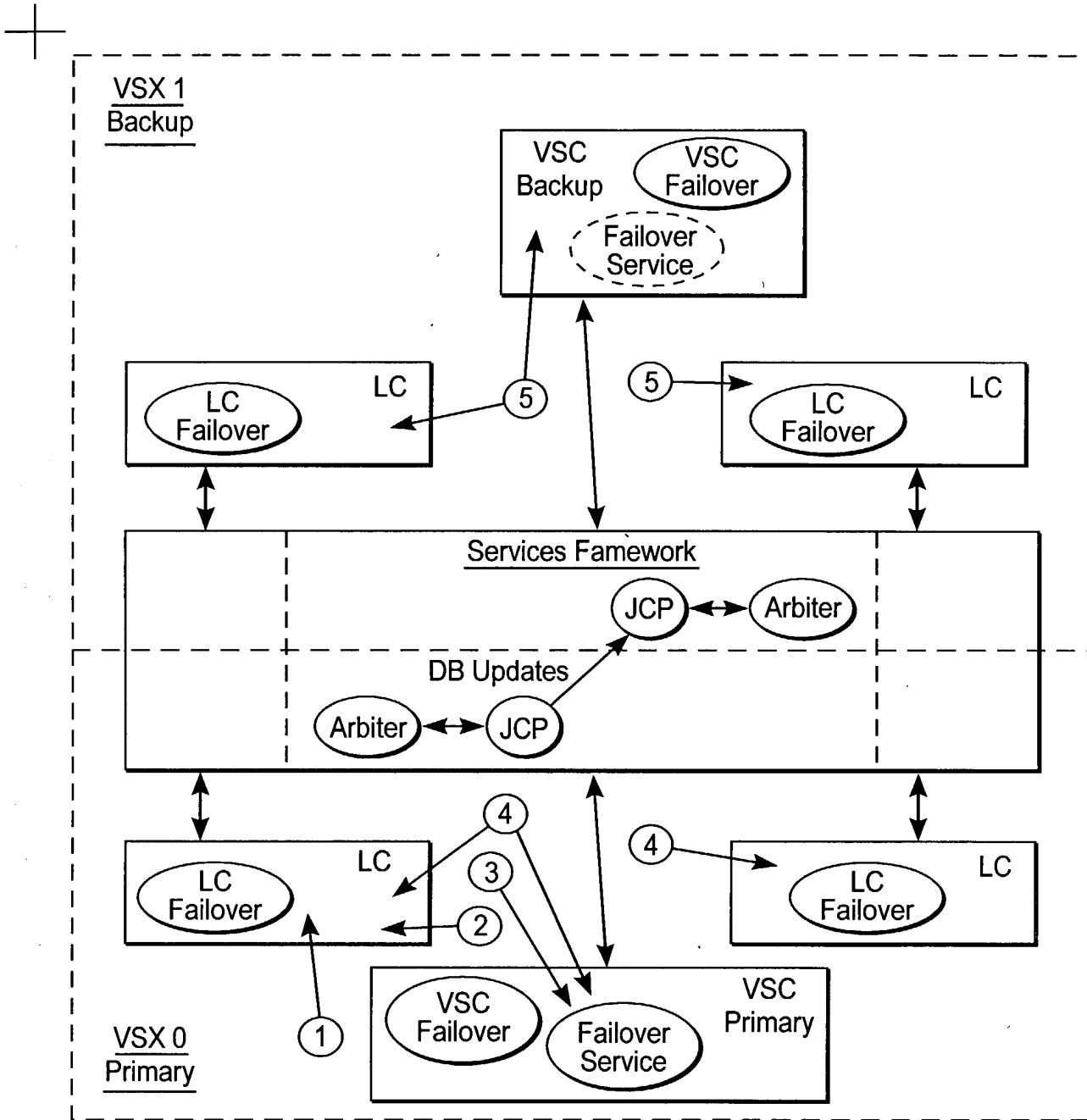


FIG. 33 IO Path Failover - LC Fails



1. FC ASIC Crashes
2. LC detects FC ASIC crash
3. Error Analysis determines IO Path fails for all devices (server or storage) on FC ASIC
4. Upstream hLUNs report CHECK CONDITION for all devices connected to failed FC Ports.
RCON and FORMAT aborted, if necessary.
5. Restart RCON and FORMAT, if necessary

FIG. 34 IO Path Failover - FC Port Fails



1. Link down on port
2. LC detects FC Port link down
3. Error Analysis determines IO Path fails for all devices (server or storage) on FC Port
4. Upstream hLUNs report CHECK CONDITION for all devices connected to FC Port.
RCON and FORMAT aborted, if necessary.
5. Restart RCON and FORMAT, if necessary

FIG. 35 IO Path Failover - Link Down

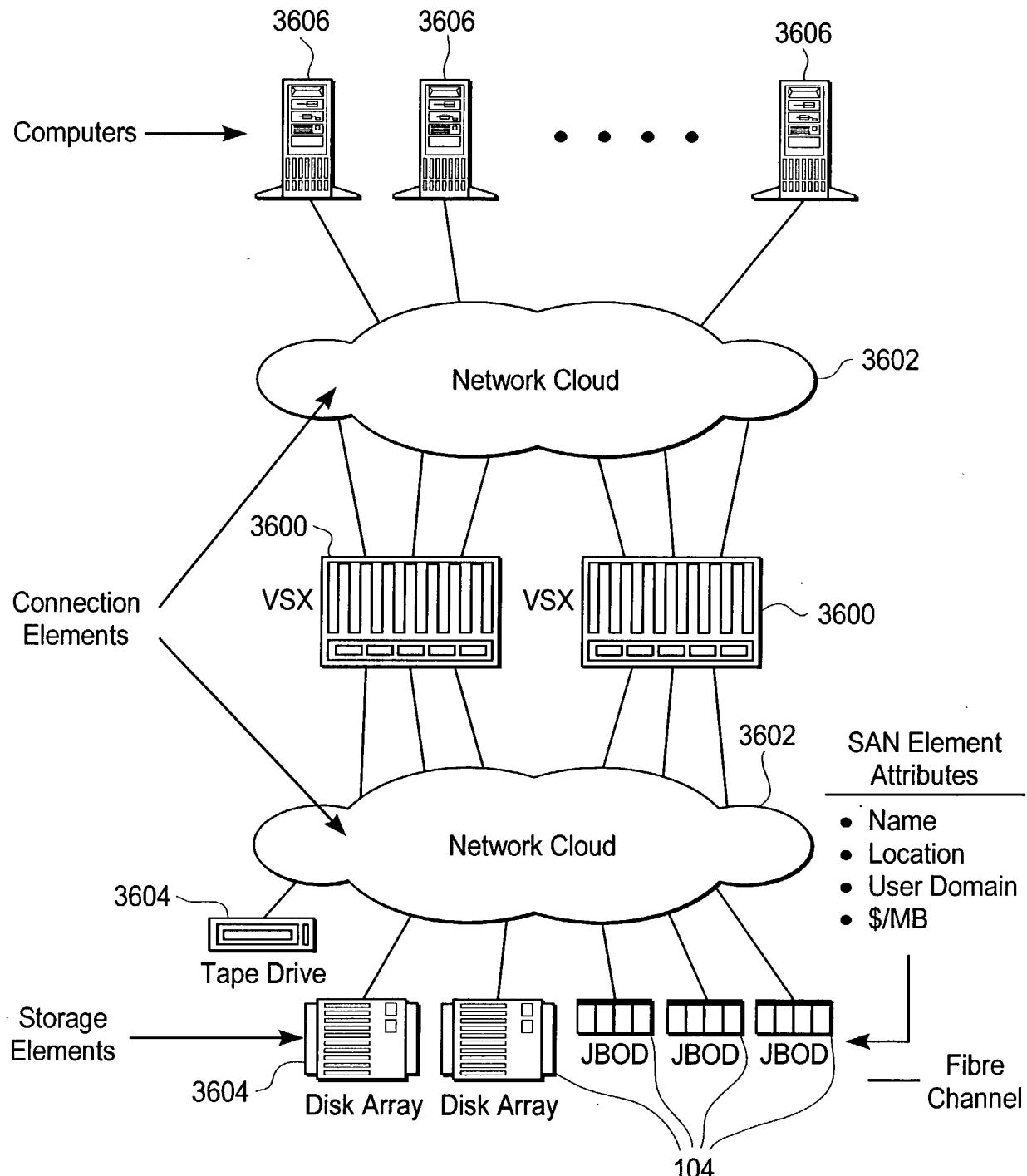


FIG. 36

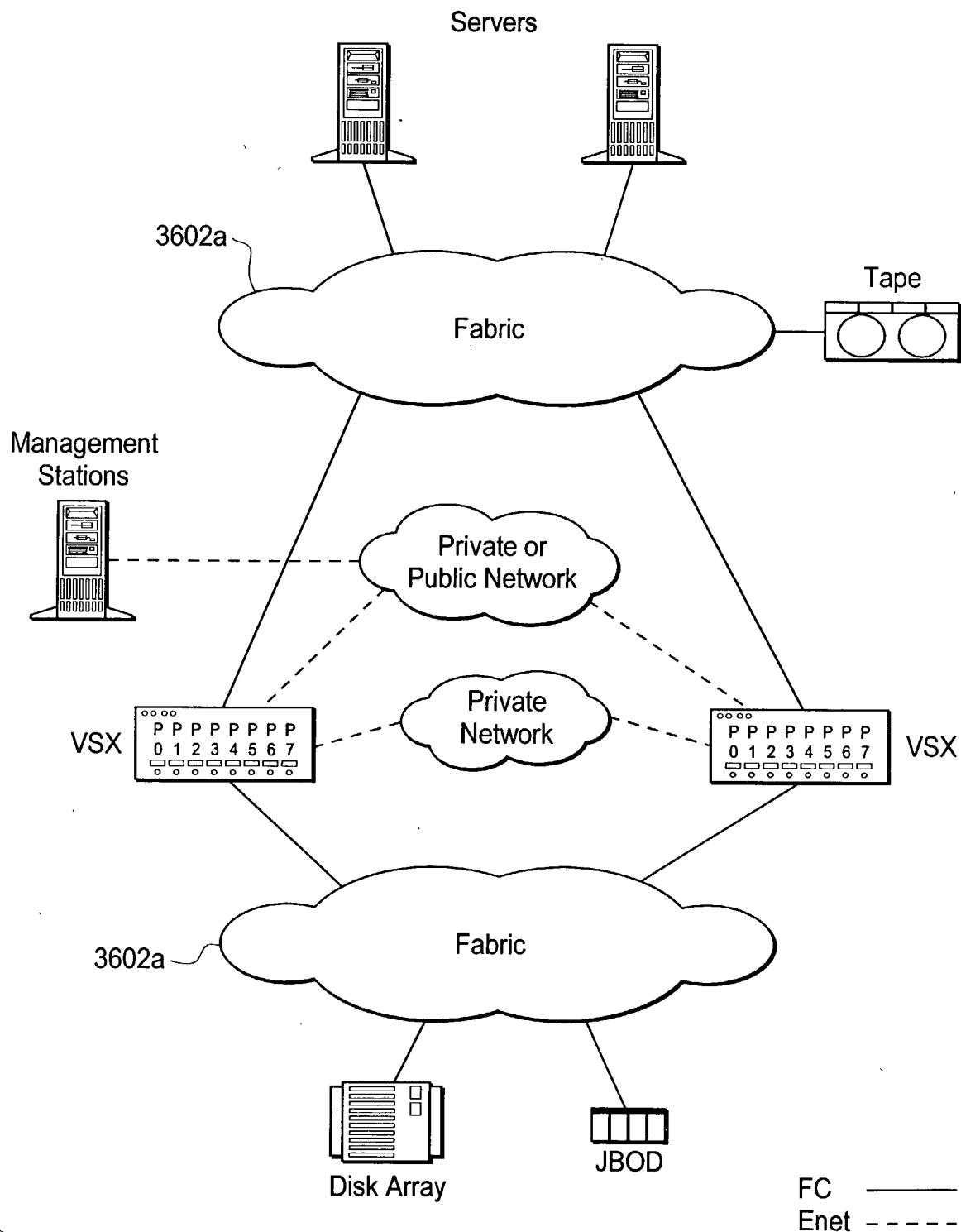


FIG. 36A Physical Setup for VSX-HA — Variation 1



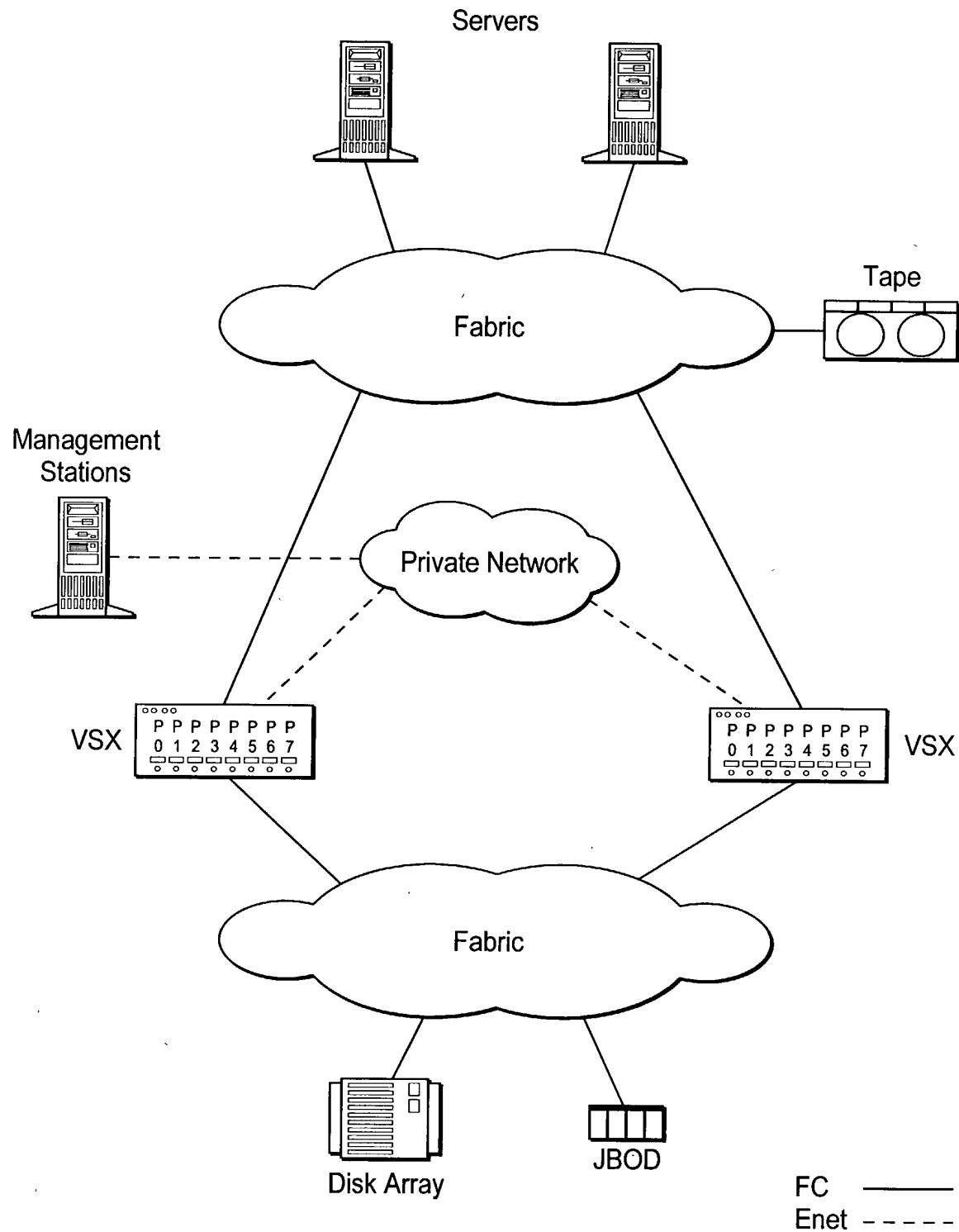


FIG. 36B Physical Setup for VSX-HA — Variation 2

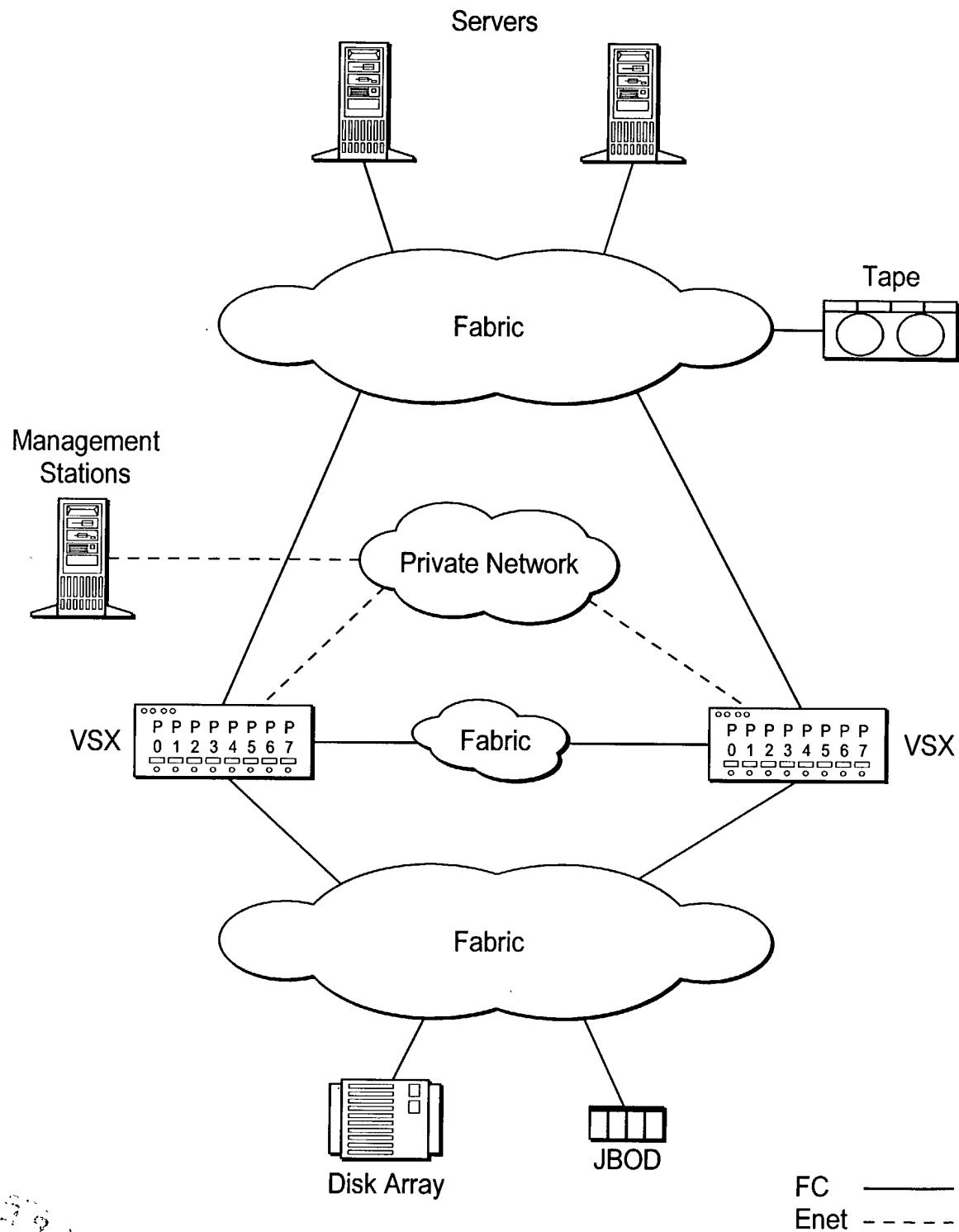


FIG. 36C Physical Setup for VSX-HA — Variation 3

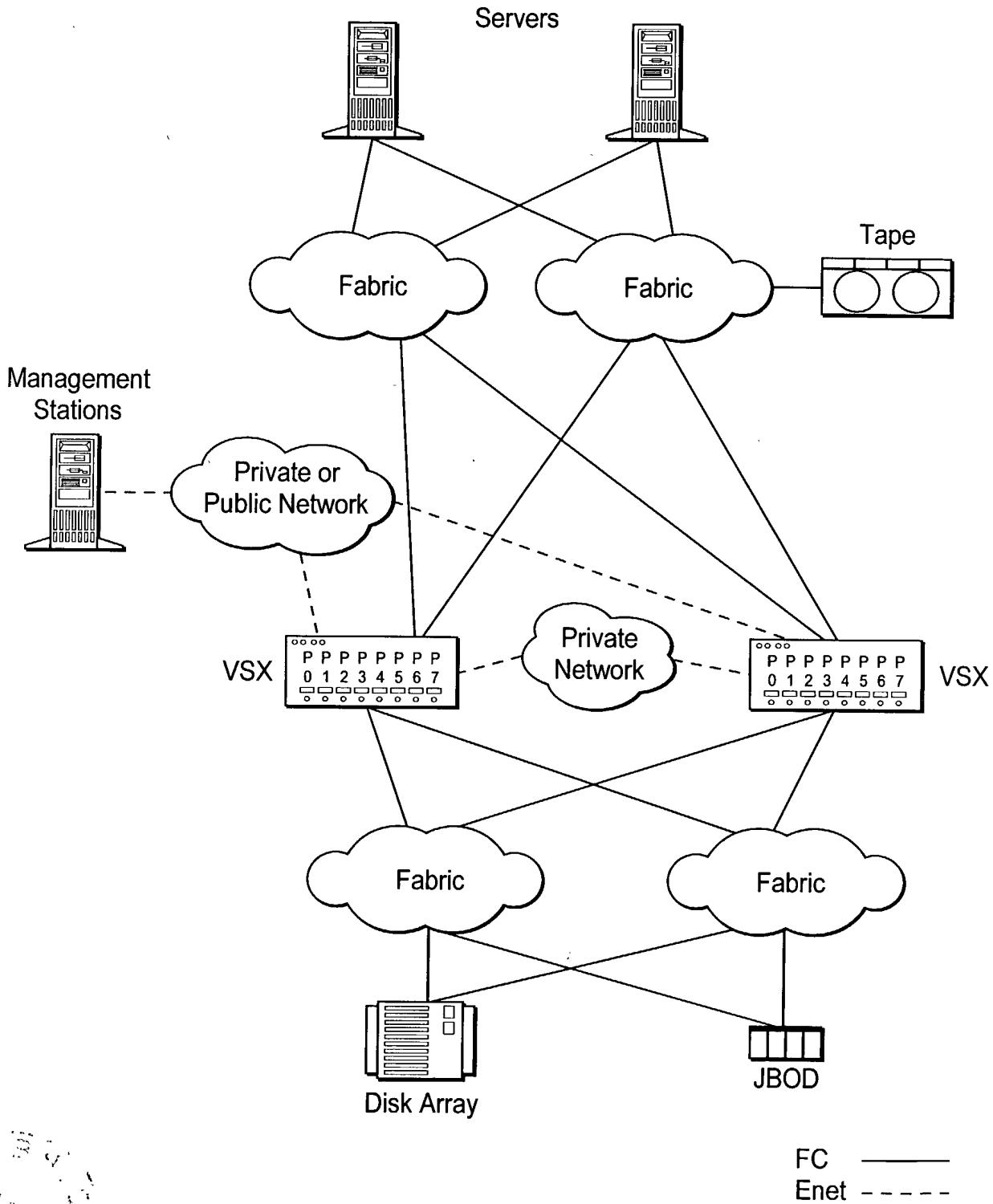


FIG. 36D Physical Setup for VSX-HA — Variation 4

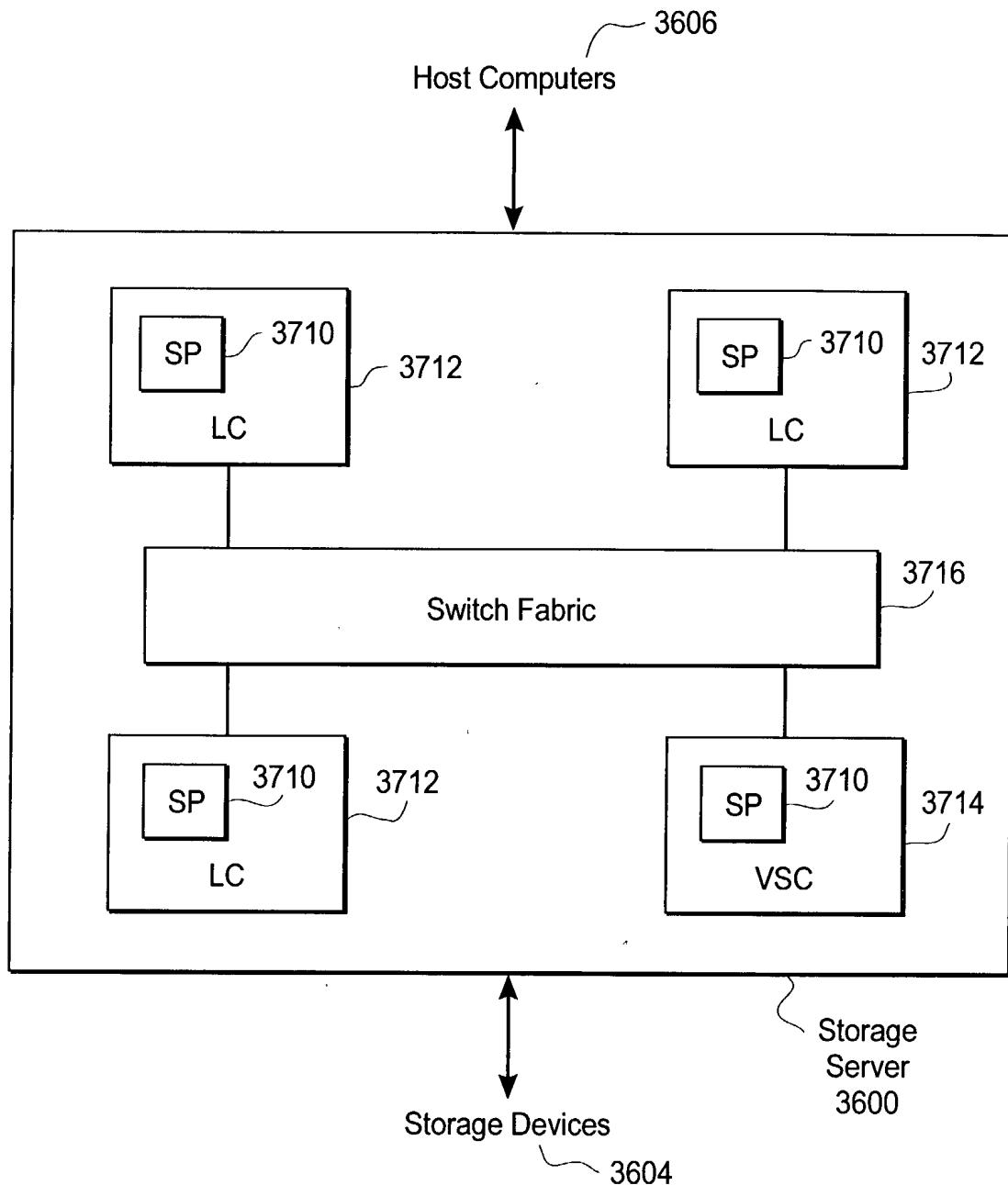


FIG. 37